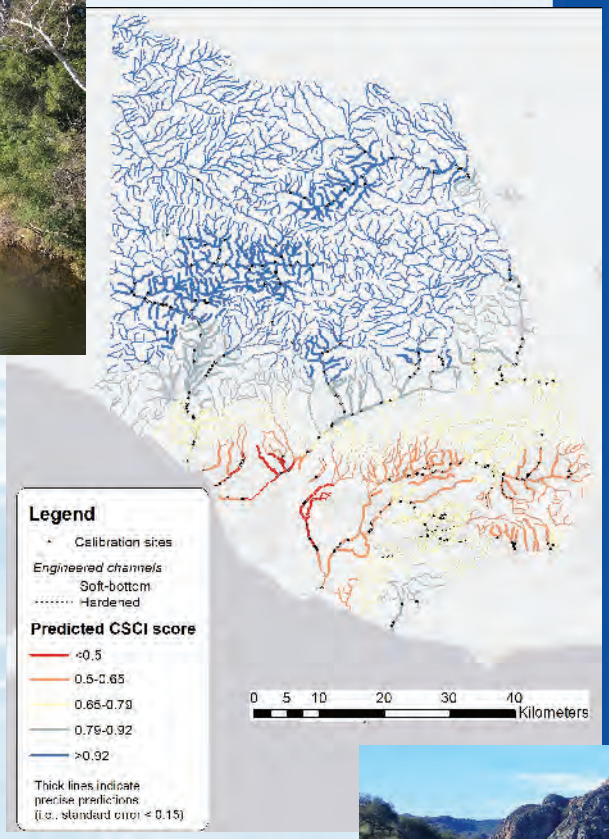
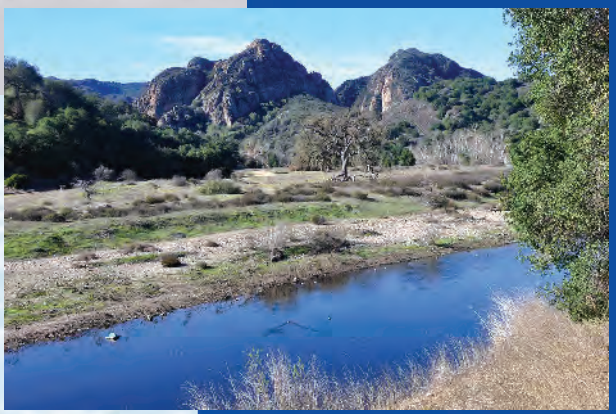
Assessing the representativeness of bioassessment samples using spatial statistical networks (SSNs) for watersheds in California: A guide for aquatic resource managers



Raphael Mazor
Abel Santana
Charlie Endris
Kevin O'Connor



Southern California Coastal Water Research Project

SCCWRP Technical Report 1143

Assessing the representativeness of bioassessment samples using spatial statistical networks (SSNs) for watersheds in California: A guide for aquatic resource managers

Raphael Mazor[1], Abel Santana[1], Charlie Endris[2], and Kevin O'Connor[2]

[1]Southern California Coastal Water Research Project, Costa Mesa, CA

[2]Central Coast Wetlands Group, Moss Landing, CA

October 2020
Technical Report 1143

ACKNOWLEDGEMENTS

This report was prepared for the California Stormwater Quality Association (CASQA).

EXECUTIVE SUMMARY

Bioassessment is a useful method for estimating the impacts of stormwater management on stream condition. However, conducting bioassessment can be difficult or costly, and managers may not be able to achieve high levels of spatial coverage in portions of the watershed they manage, forcing them to make decisions based on limited information about stream condition. Spatial statistical network (SSN) models provide a way to estimate conditions (e.g., bioassessment index scores) at unsampled locations based on their proximity to sampled locations, providing an expanse of stream condition that may not otherwise have been quantified.

We conducted four case studies to explore the value of SSNs models for stormwater managers in California: Ventura County, Los Angeles County, Santa Clara County, and Alameda County. We evaluated the importance of environmental factors (specifically, channel engineering and imperviousness in the watershed) and spatial factors (specifically, distance between sampling locations) in estimating bioassessment index scores in unsampled locations. Although we did not identify a spatial limit for extrapolations that could be applied to all streams in California, we found that SSN models are a powerful tool for understanding watershed conditions, yet substantial hurdles (such as training requirements) limit their widespread use by stormwater management agencies. We have developed a toolkit to facilitate the development of SSNs in California. This toolkit should make it easier to use SSNs in California watersheds, enhancing capacity among technical staff and consultants to create these models for bioassessment purposes, as well as other types of environmental monitoring data stormwater managers need to evaluate in stream networks.

Lessons from the case studies

Channel modification is strongly associated with differences in index scores.

- Hardened channels had low scores in all case studies, whereas scores were much more variable in soft-bottom channels. In general, extrapolating from one channel type to another should be discouraged.

Spatial models provide precise, spatially explicit estimates of stream condition.

- Spatial models were always better (i.e., more precise) than non-spatial models.

Spatial models represent good use of available data.

- Every sampled site (probabilistic or targeted) can be used to estimate overall condition.

Spatial information by itself is often sufficient for estimating condition at unsampled sites where other information is lacking.

- Land use and channel engineering information only marginally improved estimates of condition in certain circumstances.
- This superiority reflects the ability of bioassessment indicators to integrate conditions better than any proxy (even ones as integrative as “% impervious”).

Although we didn't identify a number or limit, our ability to extrapolate suggests that bioassessment samples can represent large areas with moderate precision.

- When eyeballing the maps, "high" precision estimates are limited to a few kilometers. There was no obvious difference between natural and modified channels in these limits.

Data density alone does not always lead to the most precise models.

- Other factors, like watershed hydrography and underlying variability of the data, may play a role. For example, Alameda had the fewest samples, but the best precision, while Los Angeles had a large number and high density of samples, yet the worst precision. Within a watershed, however, better sampled regions have more precise estimates than poorly sampled regions.

TABLE OF CONTENTS

INTRODUCTION

Background and purpose

As the State Water Resources Control Board (State Water Board) develops guidance for the implementation of biointegrity policies, and regional efforts (such as proposed biological objectives for the San Diego region) proceed in tandem, CASQA is supporting research that helps managers understand limits on the spatial extent of biological condition assessment based on samples from a limited number of locations. Bioassessment tools such as the California Stream Condition Index (CSCI, Mazor et al. 2016) will be used to support these efforts. As a result, there exists a need to understand the representativeness of limited numbers of reach assessments to the larger watershed condition.

CASQA previously contracted with the Southern California Coastal Water Research Project (SCCWRP) to evaluate the spatial representativeness of bioassessment monitoring data and prepared a report (Mazor et al. 2017). Upon initiation of the previous project, it became clear that more complex statistical tools were necessary and would need to be developed on a watershed basis to provide an adequate level of confidence in statistical extrapolation or interpolation. To perform the scope of services for CASQA, SCCWRP facilitated a technical advisory group and determined that it was necessary to develop spatial statistical network (SSN) models.

Under this previous project, a general spatial representative model was produced. Because of the questions regarding the application of the State's Biological Stimulatory/Integrity program on engineered channels and the prevalence of engineered channels in many MS4 permits statewide, additional evaluation of this stream type was determined to be necessary. Specifically, additional evaluation was needed to support management decisions based on the extent of likely constrained biology in engineered channels. This project supported efforts to evaluate the spatial representativeness of CSCI scoring in engineered channels or channels that have been modified substantially to support other beneficial uses.

Using the data from statewide surveys, the State Water Resources Control Board has developed a statistical model used to predict a range of likely CSCI scores based on the observed CSCI scores associated with different landscape characteristics (Beck et al. 2019). This model (called the Stream Classification and Prioritization Explorer, or SCAPE model) has recently been used by the State to support the Biological Stimulatory/Integrity program. Streams where the range of predicted scores falls below 0.79 are considered "likely constrained", and streams where the predicted range is above 0.79 are considered "likely unconstrained". This model can be used to identify constrained streams, where scores above key CSCI thresholds are unlikely to be attained. This model can help resource managers set goals that are appropriate for the constraints of their region.

These two efforts exemplify two different approaches to estimating bioassessment scores at unsampled locations. The SSN approach is based largely on spatial information: scores are estimated based on observations at nearby locations. In contrast, the landscape model approach is based purely on environmental factors (such as imperviousness, land use, or road density in the watershed); estimates are based on observations in similar environments (regardless of whether the observation is close to the unsampled location). There are both technical and philosophical differences between these approaches, and it may be best to use them in complementary fashion

where feasible. However, spatial models can be extended to make use of both spatial and environmental data. In this study, we explore efforts to use SSNs to make estimates from both spatial and environmental information. Ultimately, this model can be used to identify streams that are likely to meet managers' goals (e.g., CSCI scores above 0.79), even where monitoring data is limited. When used in combination with SCAPE model outputs, SSN models may help managers prioritize stream reaches for further investigation, restoration, or other activities.

Why would you want an SSN?

The objective of this study was to determine the spatial variability and ability to extrapolate modelled biological assessment scores (specifically, CSCI scores) in a variety of engineered channel types, and to determine their biological condition ceiling potential. Spatial statistical network (SSN) models allow for the estimation of bioassessment index scores at unsampled locations based on their proximity to sampled locations within a stream network. This modeling output can help state regulators better understand appropriate limits on the spatial extent of condition assessment based on samples from a single location, which can help inform the development of guidance for the implementation of a biological integrity policy.

Spatial statistics date back decades, particularly in the geological sciences (Peterson and Ver Hoef 2014). They can help predict locations of mineral deposits in two- or three-dimensional space based on a limited number of discrete observations. Peterson and Ver Hoef (2014) developed tools in ArcGIS and R to facilitate the creation of networks that reflect stream topology, which has accelerated the application of spatial statistics to watershed management, yet studies demonstrating applications to bioassessment or benthic macroinvertebrate data are still few in number (Frieden et al. 2014).

Spatial statistics offer a way to estimate likely biological condition at unsampled locations based on their proximity to sampled locations. Although other methods exist for estimating biological condition at unsampled locations (e.g., based on landscape alteration, Beck et al. 2019), spatial statistics are conceptually different, offering a few theoretical and practical advantages:

- Estimates based on spatial models are typically much more precise than estimates from similar non-spatial models.
- Spatial models are based on direct observation of conditions (sometimes exclusively so). In contrast, non-spatial models are based on hypothesized relationships with environmental factors (e.g., landscape alteration). Therefore, spatial models do not require assumptions about the causes of poor condition.
- Spatial models are able to show how well-sampled portions of a watershed are better understood than poorly sampled regions.

Four California case studies

We identified four case study regions (either watersheds or counties) where we expected to find large numbers of bioassessment samples, as well as information on channel engineering: Los Angeles County, Ventura County, Santa Clara County, and Alameda Creek in Alameda County. For each case study, we identified sources of channel engineering information, updated stream networks, and built models to predict CSCI scores at unsampled locations. Channel engineering information was provided as shapefiles by different local agencies. We simplified the numerous

classifications found in the original shapefiles into binary (natural vs. engineered) or three-category (natural vs. soft-bottom vs. hardened) classes. We then transferred stream channel information to the National Streams Internet (NSI), a shapefile of topologically corrected flow lines, based on the National Hydrography Dataset Plus (NHD Plus). We obtained information on watershed imperviousness from the StreamCat dataset (Hill et al. 2016) associated with nearly all stream segments in the NHD Plus (and therefore in the NSI). We then followed the steps described in the appendix to snap observed bioassessment scores with each flowline in the NSI, generate a set of prediction points (i.e., unsampled points where CSCI predictions are desired). This process results in a landscape network (LSN). The LSN is an interim product necessary for creating SSN models, which can then be used to predict CSCI scores at the prediction points.

We used the SSN package in R (Ver Hoef et al. 2014) to create three types of models that predict CSCI scores at unsampled locations:

- A ***non-spatial*** or ***purely environmental model***, where CSCI scores are predicted from environmental factors that characterize the site, such as channel form or watershed characteristics.
- A purely ***spatial model***, where CSCI scores are predicted from scores at nearby sampled locations
- A combined ***spatial environmental model***, where CSCI scores are predicted from both environmental factors and scores at nearby sampled locations.

We considered up to two types of environmental factors: channel engineering information (as either a two-category or three-category variable) or watershed imperviousness (as a continuous variable). To construct the non-spatial, purely environmental model, we identified the best combination of environmental factors based on the Akaike Information Criterion (AIC), which is a common measure for identifying parsimonious models with good prediction ability because unlike many goodness-of-fit measures (like R^2), it penalizes models for having high numbers of predictors. Low AIC values indicate good performance. However, we also looked at the root-mean square prediction error (RMSPE) to estimate the overall performance of the model; low RMSPE values indicate more precise models.

We considered up to three different spatial components in spatial models: Euclidean distance between sites (which ignores flow connectivity), “tail-down” distance (i.e., hydrologic distance between flow-connected sites in a downstream direction), and “tail-up” distance (i.e., hydrologic distance between flow-connected and flow-unconnected sites in upstream directions) (Figure 1). To construct the purely spatial model, we again used AIC to identify the best combination of spatial components.

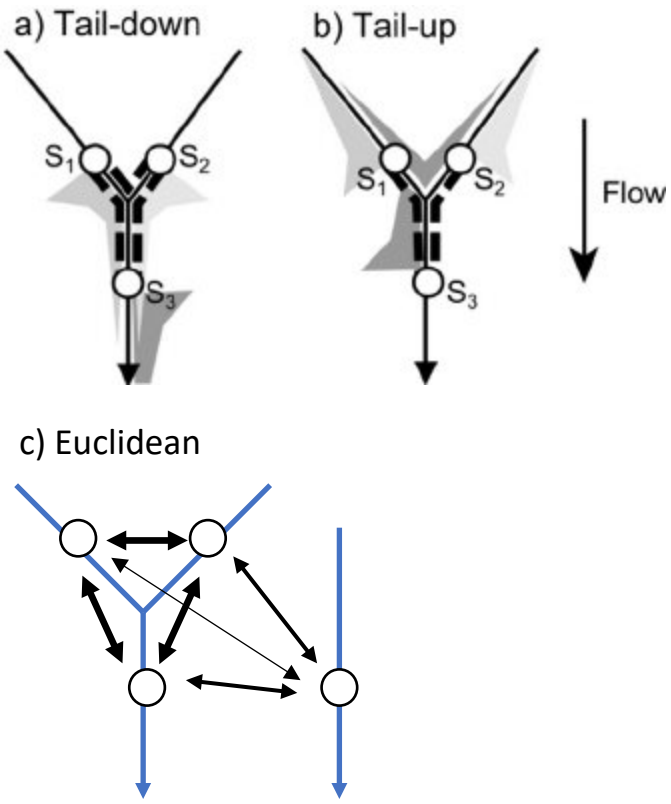


Figure 1. Diagrams of tail-down, tail-up, and Euclidean distances. The gray ribbons represent the strength of influence each site has in tail-up (a) and tail-down (b) directions. Sites S1 and S2 are hydrologically connected to S3, but not to each other. Panels A and B are from Frieden et al. 2014. Euclidean distances (c) can be calculated among all sites (even those with no hydrologic connectivity; sites closer together have stronger influences on each other (indicated by thicker black arrows).

The strength of each component can be interpreted as follows:

- If the Euclidean component is strong, sites close together tend to be similar, regardless of whether those sites are connected by flow. Thus, if you know a CSCI score at one site, sites close by will likely have the same score.
- If the tail-down component is strong, sites tend to be similar if you move in a downstream direction. Thus, if you know a CSCI score at one site, downstream sites will likely have the same score, but upstream sites might not.
- If the tail-up component is strong, sites tend to be similar if you move in an upstream direction. Thus, if you know a CSCI score at one site, upstream sites will likely have the same score, but the downstream sites might not.

We compared the predictive value of the spatial environmental model to the best purely spatial and the non-spatial models based on the AIC. We then applied the best model to a dense set of prediction points in the watershed, estimating both likely CSCI scores and the prediction standard error (which can then be used to estimate the likely range of CSCI scores at each location). These points were then used to create a map showing likely scores, highlighting regions of high and low certainty.

Summary of the data

Across the 4 case studies, we had CSCI scores from 869 locations, with higher numbers in the Southern California case studies (i.e., 363 and 282 samples for Ventura and Los Angeles Counties, respectively) than the Northern California case studies (i.e., 118 and 106 samples for Santa Clara and Alameda Counties, respectively). In Los Angeles, Santa Clara, and Alameda, samples from natural channels outnumbered engineered channels, while in Ventura, samples from engineered channels were more common.

In all case studies, scores in hardened channels were typically lower than in soft-bottom or natural channels, although low scores were observed in all channel types. Ranges were relatively homogenous in hardened channels, with medians ranging from 0.40 to 0.60, and 90th percentiles (an estimate of the upper limit of scores in each channel type) ranging from 0.52 to 0.73 (Table 1). In contrast, ranges were much larger in natural and soft-bottom channels. Notably, values for soft-bottom channels varied considerably from case to case, with relatively low scores observed in Alameda County and much higher scores in Ventura County (Figure 2). Thus, the soft-bottom classification may reflect a mix of channel types that varies from case study to case study.

Table 1. Summary of CSCI scores used in the study. n = Number of unique samples.

County	Channel type	n	10th percentile CSCI	Median CSCI	90th percentile CSCI
Alameda	Natural	74	0.43	0.70	1.07
	Soft	10	0.23	0.34	0.40
	Hardened	22	0.20	0.40	0.52
Los Angeles	Natural	183	0.60	0.92	1.18
	Soft	19	0.26	0.48	0.71
	Hardened	80	0.32	0.60	0.73
Santa Clara	Natural	83	0.45	0.74	1.11
	Soft	24	0.41	0.53	0.70
	Hardened	11	0.25	0.42	0.64
Ventura	Natural	129	0.74	0.99	1.13
	Soft	219	0.49	0.72	1.02
	Hardened	15	0.39	0.50	0.67

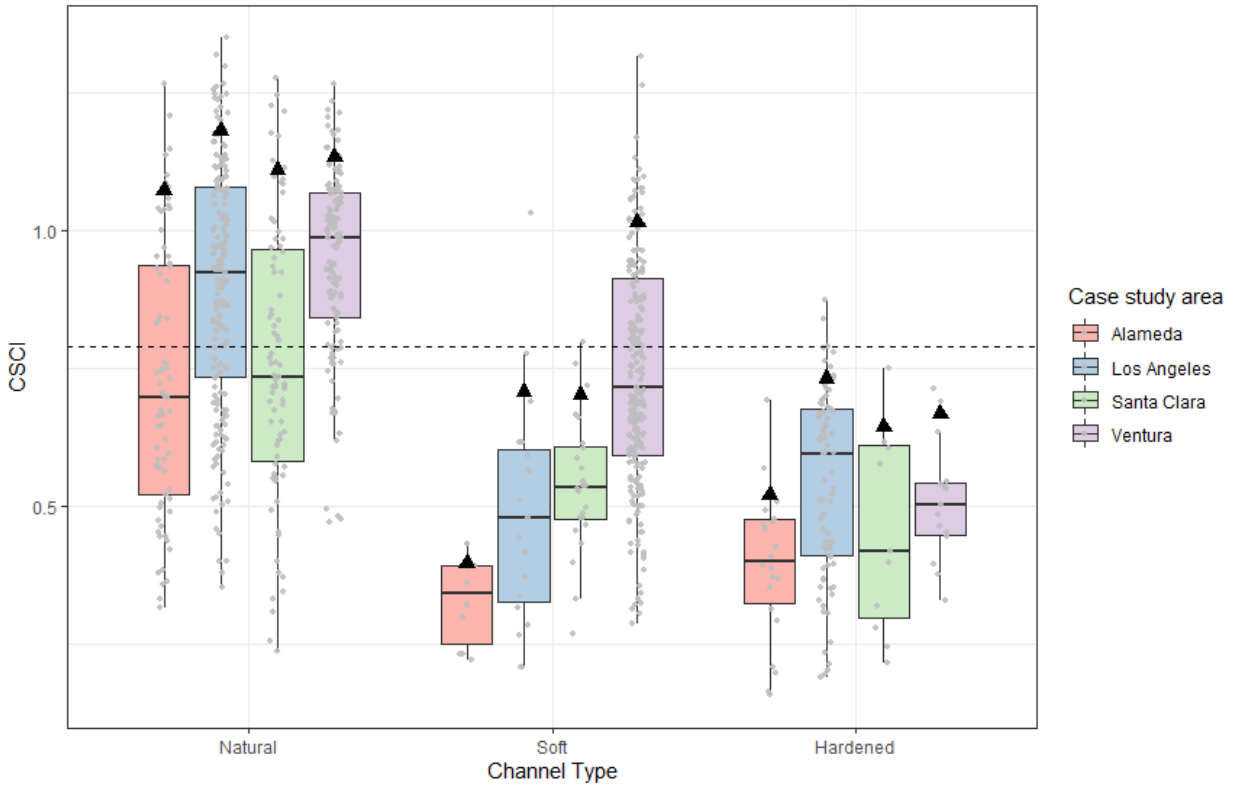


Figure 2. Range of CSCI scores in different channel types in the four case studies. Gray dots indicate individual samples. Large black triangles indicate the upper limit (i.e., the 90th percentile) of scores in each channel type for each case study.

Ventura County

A shapefile showing stormwater infrastructure was provided by Ventura County Watershed Protection District. This shapefile contained channel classifications used by County staff for operations and maintenance indicated in the "OM2" field of the shapefile attribute table. Definitions of some of these classes was self-evident (e.g., "concrete lined channel"), but others were less clear (e.g., "redline") and not readily available; based on discussions with County staff and review of aerial imagery, OM2 designations were classified as natural or engineered; engineered features were further classified as hardened or soft-bottomed (Table 2). In the absence of contradictory information, we assumed unclassified channels were natural, and that unspecified but clearly engineered channels were soft-bottom. Although we conducted limited validation of these assignments using Google Earth, it is not possible to completely verify the accuracy of these classifications and how well they represent present-day conditions; as with all the case studies, it is likely that some inaccuracies (including misclassifications, exclusions of streams, and inclusions of flowlines that do not represent aquatic habitats) affect the data.

Table 2. Classification of channel types provided by the Ventra County Watershed Protection District

Natural streams	Engineered – hard-bottomed streams	Engineered – soft-bottomed streams
Blueline Connector Mining area NHD Natural stream	Concrete lined channel Conduits	Bank protection Dam/Debris Basin/NPDES Basin Improved unlined channel Flowline Invert stabilization Levee Other Redline Transition V-Ditch

The stormwater infrastructure shapefile attributes were then transferred to the National Streams Internet (NSI) flowlines, which are required for SSN modeling. Flowlines in the NSI not found in the stormwater infrastructure shapefile were designated as soft-bottom where local riparian imperviousness exceeded 15%; otherwise, they were designated as natural channels. The resulting network showed that Ventura County contained 4,595 stream-km, 77% of which are natural, 22% are soft-bottom, and 1% are hard-bottom.

A landscape network (LSN) was created from the NSI and modified to correct topological errors (typically by deleting isolated networks where the errors were found). The LSN is an interim product necessary for creating SSN models.

The best non-spatial model for predicting CSCI scores was based on two factors: percent imperviousness in the watershed, plus a binary variable indicating whether the channel was engineered or natural (Table 3). Subdividing the engineered channels based on bottom-hardening did not improve the model; in fact, that model had larger error than any other model evaluated — perhaps due to the small number of samples in hard-bottom channels. Adding a spatial component greatly improved performance. A purely spatial model had the best performance of all, indicating that in this intensively sampled region, information about channel type or watershed imperviousness add no additional value to predicting CSCI scores when nearby scores are known. The RMSPE of the selected (pure spatial) model was 0.13, which is small relative to the variability in CSCI scores among reference sites (i.e., 0.16; Mazor et al. 2016). This low error suggests that the model may be suitable for predicting which streams are in likely altered condition (i.e., CSCI scores < 0.79, or 0.21 points below the expected value of 1.00), but may not be suitable for detecting streams in possibly altered condition (i.e., CSCI scores < 0.92, or 0.08 points below the expected value of 1.00).

Table 3. Summary of models for the Ventura County case study. Each X indicates if an environmental factor or spatial component was included in the model. AIC: Akaike Information Criterion. RMSPE: Root-mean square prediction error. Highlighting indicates the final selected model.

Environmental factors			Spatial components				
Imperviousness	Natural vs. engineered	Natural vs. Hardened vs. Soft-bottom	Tail-up	Tail-down	Euclidean	AIC	RMSPE
Pure spatial (only top-performing model shown)							
				X	X	-382	0.13
Spatial and environmental							
X	X			X	X	-366	0.13
Non-spatial (pure environmental)							
X	X					-211	0.18
X		X				-205	0.24
		X				-153	0.19
X						-150	0.19
	X					-141	0.20

We created a set of prediction-points located every ~250 m along the stream network, and used the purely spatial model to predict CSCI scores (along with prediction errors) at each point (Figure 3). We then used these points to update the Ventura stream network to create a map showing likely scores throughout the county.

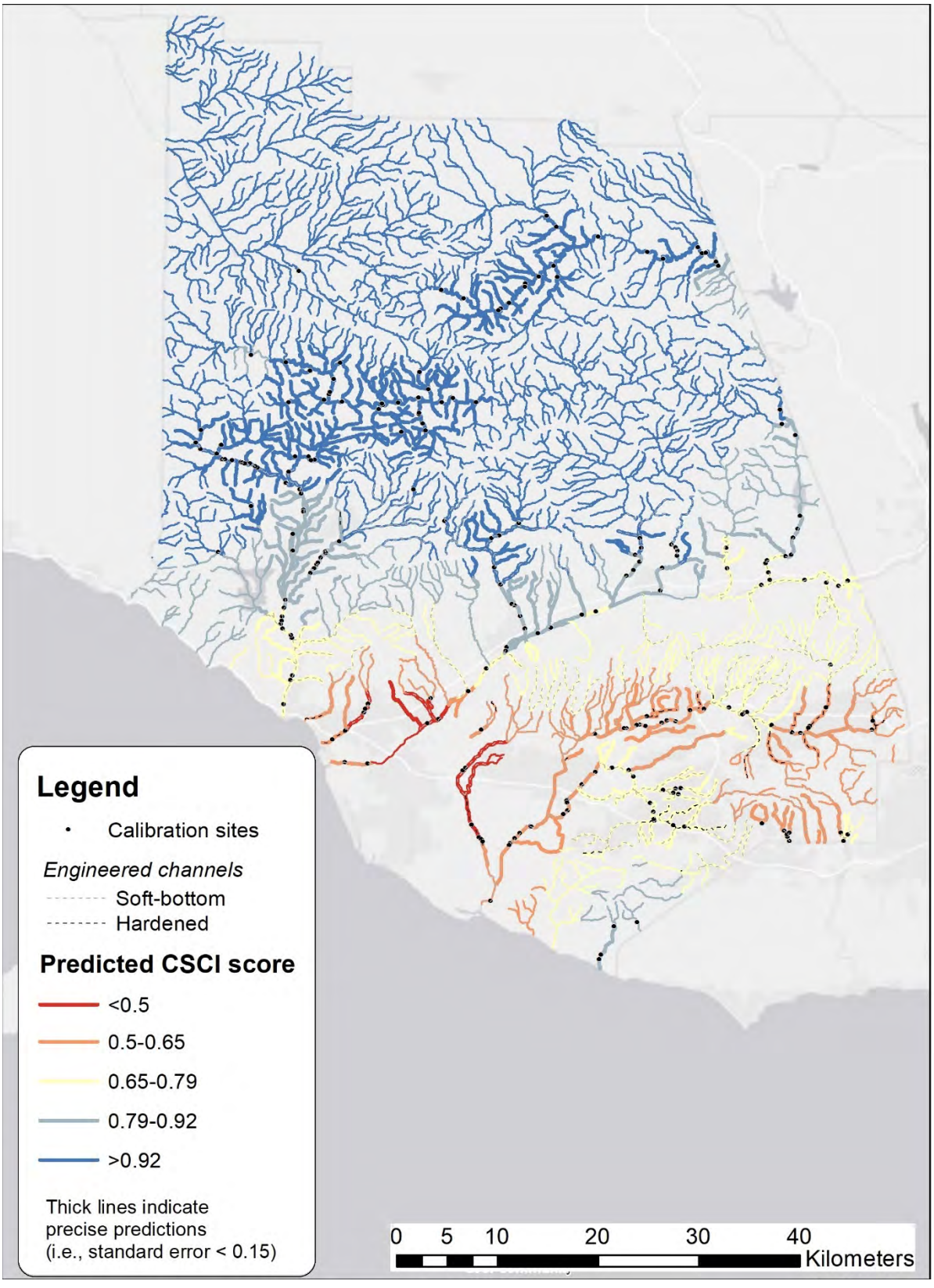


Figure 3. Predicted CSCI scores in Ventura County streams.

Overall, 74% of stream-miles were predicted to have CSCI scores indicative of reference conditions (i.e., ≥ 0.79). These streams with high predicted CSCI scores were concentrated in the undeveloped, mountainous interior portions of the county. Although these results were expected, it is nonetheless notable that these predictions are based strictly on spatial proximity to sampled sites, and not on land use or other environmental factors known to influence CSCI scores. High precision estimates (i.e., prediction standard error < 0.15) were possible for 27% of stream-miles, although 91% of estimates achieved at least moderate precision (standard error < 0.2) (Figure 4). Estimates for engineered channels (particularly hardened channels) were frequently more precise than estimates for natural channels (Table 4).

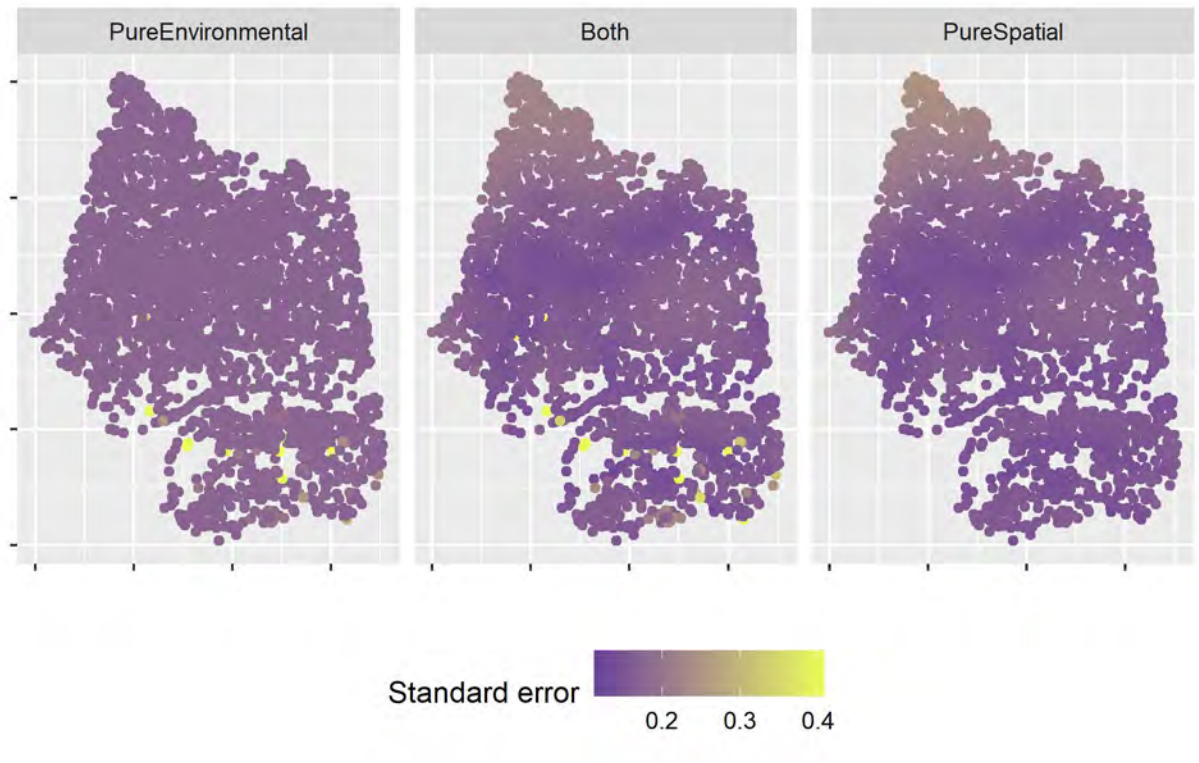


Figure 4. Prediction standard errors for the top models in Ventura County.

Table 4. Percent of stream-miles with high, medium and low precision (based on ranges of standard error shown in parentheses) in Ventura County.

Channel type	High (<0.15)	Medium (0.15 to 0.20)	Low (>0.20)
Natural	19	69	11
Engineered	46	54	0
-Hardened	60	40	0
-Soft	45	55	0

Los Angeles County

Los Angeles County Public Works provided a shapefile of modified channels comparable to the layer provided by Ventura County, with the same categories of channel types identified (Table 2). The shapefile attribute table contained a column designating hardened and soft-bottom channels, which we used as the primary classification system for analysis. Streams missing from that data set were presumed to be natural, unless imperviousness within 100 m of the stream-line exceeded 15%. The resulting network showed that Los Angeles County contained 5,745 stream-km, 84% of which are natural, 3% are soft-bottom, and 13% are fully hardened. The NSI did not contain errors requiring further modification for creating an LSN.

The best non-spatial model for predicting CSCI scores was based on the same two factors identified in Ventura County: percent imperviousness in the watershed, plus a binary variable indicating whether the channel was engineered or natural (Table 5). In contrast with Ventura County, a purely spatial model was only slightly better than a model on both spatial and environmental factors, perhaps due to the lower density of sampling locations and higher variability in scores. Furthermore, the purely spatial model for Los Angeles County had particularly poor precision (i.e., standard error > 0.3) in the sparsely sampled Antelope Valley. Therefore, the model based on both factors was used for further analysis. The RMSPE of the selected (spatial and environmental) model was 0.26, which is twice the error of the Ventura County model. This low precision suggests that the model may not be suitable for predicting which streams are in likely altered condition, but may be better suited for predicting which streams are very likely altered (i.e., CSCI < 0.63, or 0.37 points below the expected value of 1.00).

Table 5. Summary of models for the Los Angeles County case study. Each X indicates if an environmental factor or spatial component was included in the model. AIC: Akaike Information Criterion. RMSPE: Root-mean square prediction error. Highlighting indicates the selected model.

Environmental factors			Spatial components				
Imperviousness	Natural vs. engineered	Natural vs. Hardened vs. Soft-bottom	Tail-up	Tail-down	Euclidean	AIC	RMSPE
Pure spatial (only top-performing model shown)							
				X	X	82	0.262
Spatial and environmental							
X	X			X		85	0.260
Non-spatial (pure environmental)							
X	X					106	0.277
X		X				118	0.278
	X					130	0.295
		X				133	0.296
X						136	0.295

We created a set of prediction-points located every ~250 m along the stream network, and used the environmental + spatial model to predict CSCI scores (along with prediction interval) at each point (Figure 5). We then used these points to update the Los Angeles County stream network to create a map showing likely scores throughout the county.

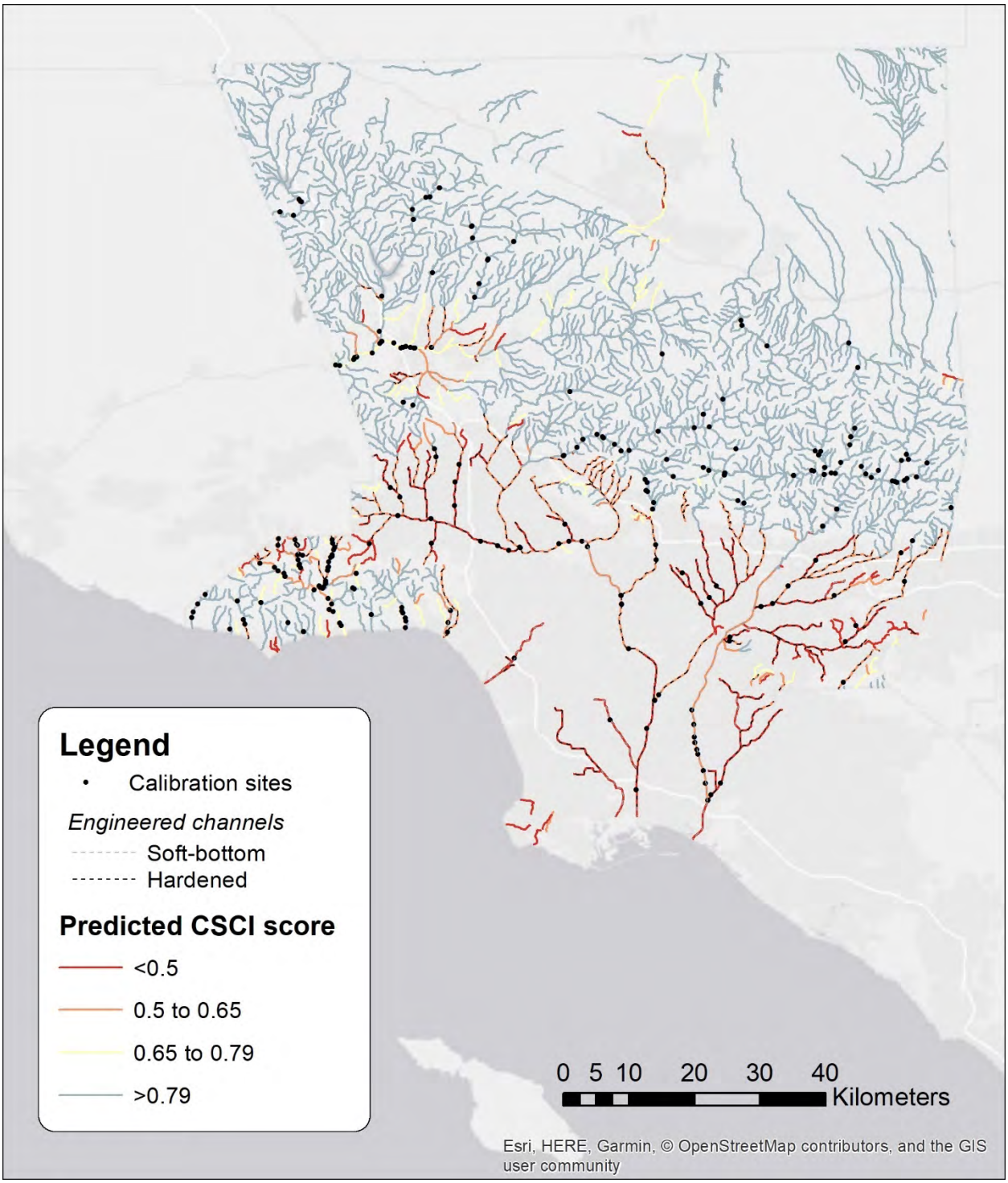


Figure 5. Predicted CSCI scores in Los Angeles County streams.

Overall, 77% of stream-miles were predicted to have CSCI scores indicative of reference conditions (i.e., ≥ 0.79). These streams were concentrated in the undeveloped, mountainous interior portions of the county. Compared with Ventura County, precision of the model in Los Angeles county was poor, with no estimates achieving a standard error as low as 0.28 (Figure 6), and all predictions were classified as having low precision (i.e., standard error > 0.20; Table 6).

This poor precision may be due to the complex topography of the landscape, the low density of samples in the desert portions, or to other factors that require further exploration.

Figure 6. Prediction standard errors for the top models in Los Angeles County. Note that the color scale differs greatly from the scale in Figure 4.

Table 6. Percent of stream-miles with high, medium and low precision (based on ranges of standard error shown in parentheses) in Los Angeles County.

Channel type	High (< 0.15)	Medium (0.15 to 0.20)	Low (> 0.20)
Natural	0	0	100
Engineered	0	0	100
-Hardened	0	0	100
-Soft	0	0	100

Santa Clara County

Consultants representing Santa Clara County (specifically, Nick Zigler of EOA, Inc.) provided us with two spatial datasets of engineered stream channels (Table 7). After comparing the shapefile to aerial imagery, we chose to use the file "exempt_channel.shp", created by The Habitat Restoration Group for the Riparian Corridor Policy Study (1995), due to its better classification and accuracy of engineered channels in urban zones. All channel types were aggregated and re-classified as Natural/Engineered and Natural/Soft/Hardened and then transferred to the NSI dataset ("Coyote" HUC8 watershed). NSI Channels that did not overlap with the engineered channels were assumed to be natural, and verified using NLCD Impervious layers. The resulting network showed that Santa Clara County contained 1185 stream-km, 84% of which are natural, 10% are soft-bottom, and 6% are hard-bottom. The NSI did not contain errors requiring further modification for creating an LSN.

Table 7. Classification of channel types provided by Santa Clara County.

Natural Streams	Engineered – hard-bottomed streams	Engineered – soft-bottomed streams
Natural Unmodified	Arch Culvert	Earth Levees
	Articulated Concrete Blocks	Excavated Earth
	Box Culvert	Flood Walls
	Bridge	Modified Flood Plain
	Bypass Channel	
	Concrete (Bottom)	
	Gabion (Sides and Bottom)	
	Pipe Culvert	
	Rock Lined (Sides and Bottom)	
	Sack Concrete	
	Trapezoidal Concrete	
	U-Frame Concrete	

The best non-spatial model for predicting CSCI scores was based on the same two factors identified in the other two counties: percent imperviousness in the watershed, plus a binary variable indicating whether the channel was engineered or natural (Table 8). As with Ventura County, a pure-spatial model was better than the spatial + environmental factors model in Santa Clara County. However, the Santa Clara model included only a Euclidean component (i.e., overland distance; Figure 1), whereas both Los Angeles and Ventura Counties included a tail-down component. Therefore, hydrologic connectivity (which would be reflected by the presence of a tail-up or tail-down component) may not be necessary for predicting CSCI scores from nearby samples, perhaps due to the fact that this region consists largely of small, disconnected stream networks, whereas the others are characterized by a few large stream networks. We used the pure spatial model for further analyses. The RMSPE of the selected (pure spatial) model was 0.17, which is comparable to the variability in CSCI scores among reference sites (i.e., 0.16; Mazor et al. 2016). As with the Ventura County model, this low error suggests that the model may be suitable for predicting which streams are in likely altered condition (i.e., CSCI scores < 0.79, or 0.21 points below the expected value of 1.00), but may not be suitable for detecting

streams in possibly altered condition (i.e., CSCI scores < 0.92, or 0.08 points below the expected value of 1.00).

Table 8. Summary of models for the Santa Clara County case study. Each X indicates if an environmental factor or spatial component was included in the model. AIC: Akaike Information Criterion. RMSPE: Root-mean square prediction error. Highlighting indicates the selected model.

Environmental factors			Spatial components				
Imperviousness	Natural vs. engineered	Natural vs. Hardened vs. Soft-bottom	Tail-up	Tail-down	Euclidean	AIC	RMSPE
Pure spatial (only top-performing model shown)							
					X	-44	0.167
Spatial and environmental							
X	X				X	-40	0.167
Non-spatial (pure environmental)							
X	X					-21	0.202
X		X				-11	0.204
X						-2	0.229
	X					-2	0.231
		X				-1	0.231

We created a set of prediction-points located every ~250 m along the stream network, and used the pure spatial model to predict CSCI scores (along with prediction errors) at each point (Figure 7). We then used these points to update the Santa Clara County stream network to create a map showing likely scores throughout the county.

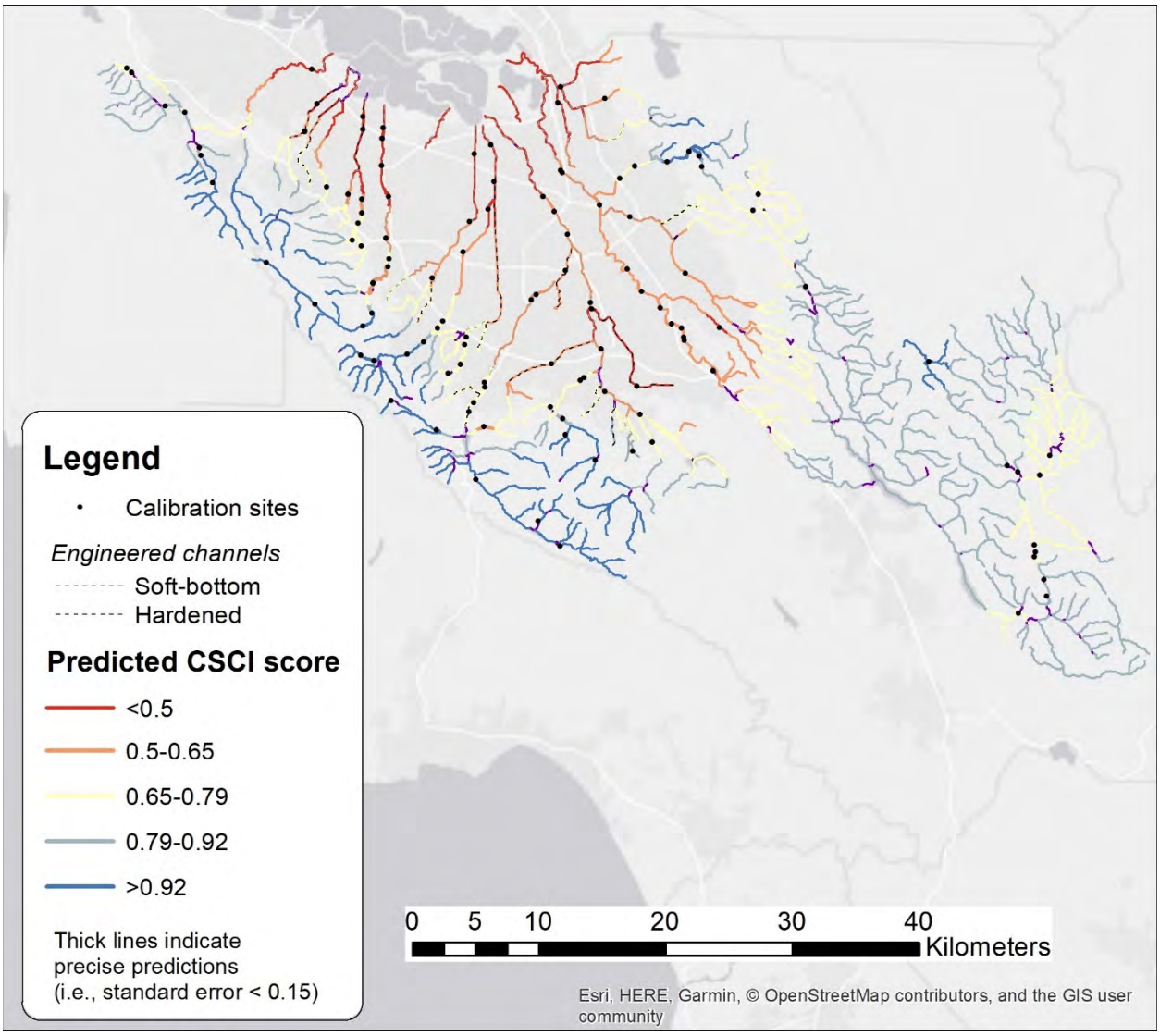


Figure 7. Predicted CSCI scores in Santa Clara County streams.

Whereas the pure-environmental model had moderate precision throughout the region, both the pure-spatial model and the spatial + environmental models had higher precision overall, with pockets of poor precision in the southeastern portion of the region (where sample density was relatively low). Although this pattern was evident in Los Angeles and Ventura Counties, it is far more striking in this county. Good conditions (i.e., CSCI $\geq$ 0.79) were predicted in the undeveloped hillsides surrounding the urban core. Poor conditions were predicted in the baylands, as well as in the headwaters of Coyote Creek east of the City of Morgan Hill. Nearly half of the watershed (i.e., 47% of stream-miles) were predicted to have CSCI scores below 0.79, indicated that poor conditions are more pervasive here than in Ventura or Los Angeles counties.

Precision was much better than in the Los Angeles case study, but not quite as good as Ventura; precise estimates (standard error < 0.15) were only achieved for 5% of the watershed, but moderate precision (standard error between 0.15 and 0.2) was achieved for 34%. Precision was poorest in the southeastern portions of the county (Figure 8). In Santa Clara County, engineered

channels were more precisely estimated than natural channels, although high precision estimates (i.e., standard error < 0.15) still relatively uncommon (i.e., only 10% of engineered channels; Table 9).

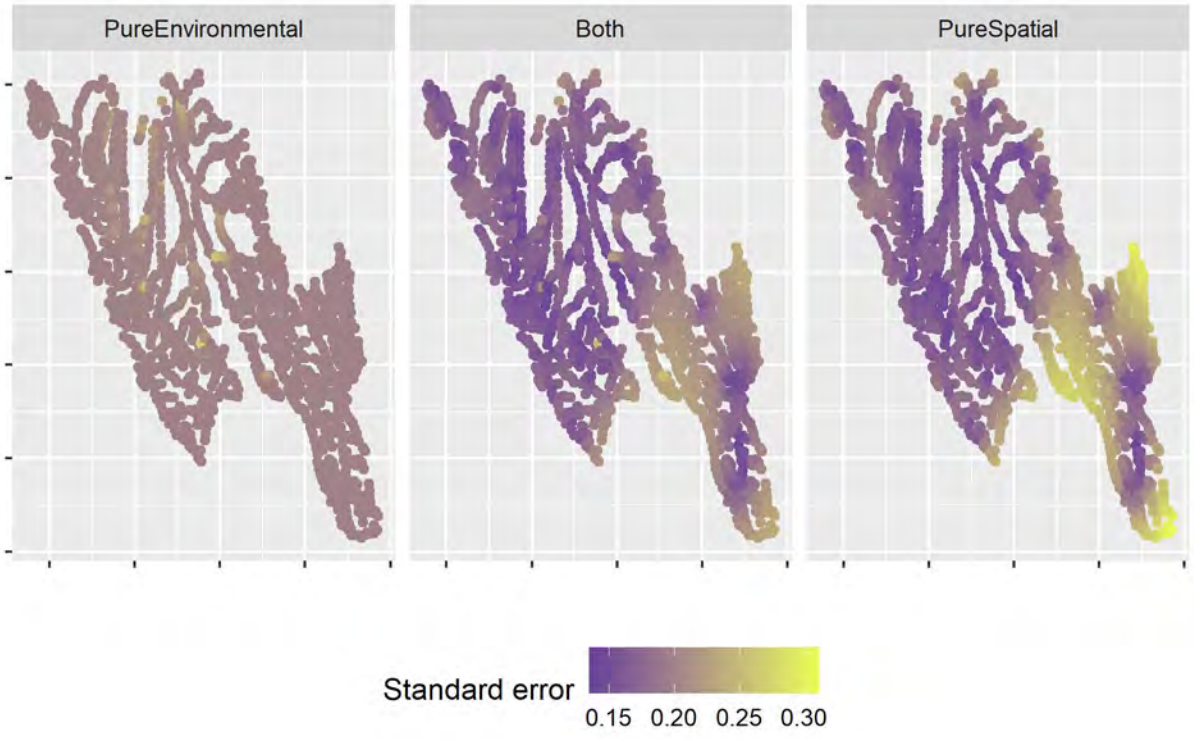


Figure 8. Prediction standard errors for the top models in Santa Clara County.

Table 9. Percent of stream-miles with high, medium and low precision (based on ranges of standard error shown in parentheses) in Santa Clara County.

Channel type	High (< 0.15)	Medium (0.15 to 0.20)	Low (> 0.20)
Natural	5	67	28
Engineered	10	90	0
Hardened	7	93	0
Soft	11	88	1

Alameda County

The Bay Area Aquatic Resource Inventory (BAARI) stream dataset, provided a shapefile with engineered channel types for Alameda County ("San Francisco Bay" HUC8 watershed; Table 10). All fluvial engineered channel types (i.e., ditch, subsurface drainage, and engineered channel) were aggregated and re-classified as Natural/Engineered and Natural/Soft/Hardened and then transferred to the NSI dataset. NSI Channels that did not overlap with the engineered channels were assumed to be natural, and verified using NLCD Impervious layers. The resulting network showed that Alameda County contained 1681 stream-km, 90% of which are natural, 3% are soft-bottom, and 7% are hard-bottom. The NSI did not contain errors requiring further modification for creating an LSN. The aggregated CSCI scores from 118 samples in Santa Clara County ranged from 0.22 to 1.28, with a mean score of 0.69.

Table 10. Classification of channel types in Alameda County.

Natural Streams	Engineered – hard-bottomed streams	Engineered – soft-bottomed streams
Fluvial Channel (FC)	Fluvial Subsurface Drainage (FSD) Fluvial Engineered Channel (FEC)	Fluvial Ditch (FD)

As with the three other counties, the best non-spatial model for predicting CSCI scores was based on percent imperviousness in the watershed, plus a binary variable indicating whether the channel was engineered or natural (Table 11). Adding a spatial component to the environmental model (specifically, a Euclidean and a tail-up component; see Figure 1) greatly improved model performance. A pure spatial model had slightly higher error rates than the combined environmental + spatial model, so the combined model was used for further analysis, as it had the highest precision in most regions of the county. The RMSPE of the selected (spatial and environmental) model was 0.14, which is lower than the variability in CSCI scores among reference sites (i.e., 0.16; Mazor et al. 2016). As with the Ventura and Santa Clara County models, this low error suggests that the model may be suitable for predicting which streams are in likely altered condition (i.e., CSCI scores <0.79, or 0.21 points below the expected value of 1.00), but may not be suitable for detecting streams in possibly altered condition (i.e., CSCI scores < 0.92, or 0.08 points below the expected value of 1.00).

Table 11. Summary of models for the Alameda County case study. Each X indicates if an environmental factor or spatial component was included in the model. AIC: Akaike Information Criterion. RMSPE: Root-mean square prediction error. Highlighting indicates the selected model.

Environmental factors			Spatial components				
Imperviousness	Natural vs. engineered	Natural vs. Hardened vs. Soft-bottom	Tail-up	Tail-down	Euclidean	AIC	RMSPE
Pure spatial (only top-performing model shown)							
			X		X	-63	0.150
Spatial and environmental							
X	X		X		X	-60	0.139
Non-spatial (pure environmental)							
X	X					-26	0.189
X		X				-17	0.188
	X					-12	0.219
		X				-10	0.219
X						-1	0.225

We created a set of prediction-points located every ~250 m along the stream network, and used the spatial + environmental factor model to predict CSCI scores (along with prediction errors) at each point (Figure 9). We then used these points to update the Alameda County stream network to create a map showing likely scores throughout the county. Good conditions (i.e., CSCI $\geq$ 0.79) were predicted in the undeveloped hillsides surrounding the urban baylands, the Livermore Valley, and Arroyo Mocho.

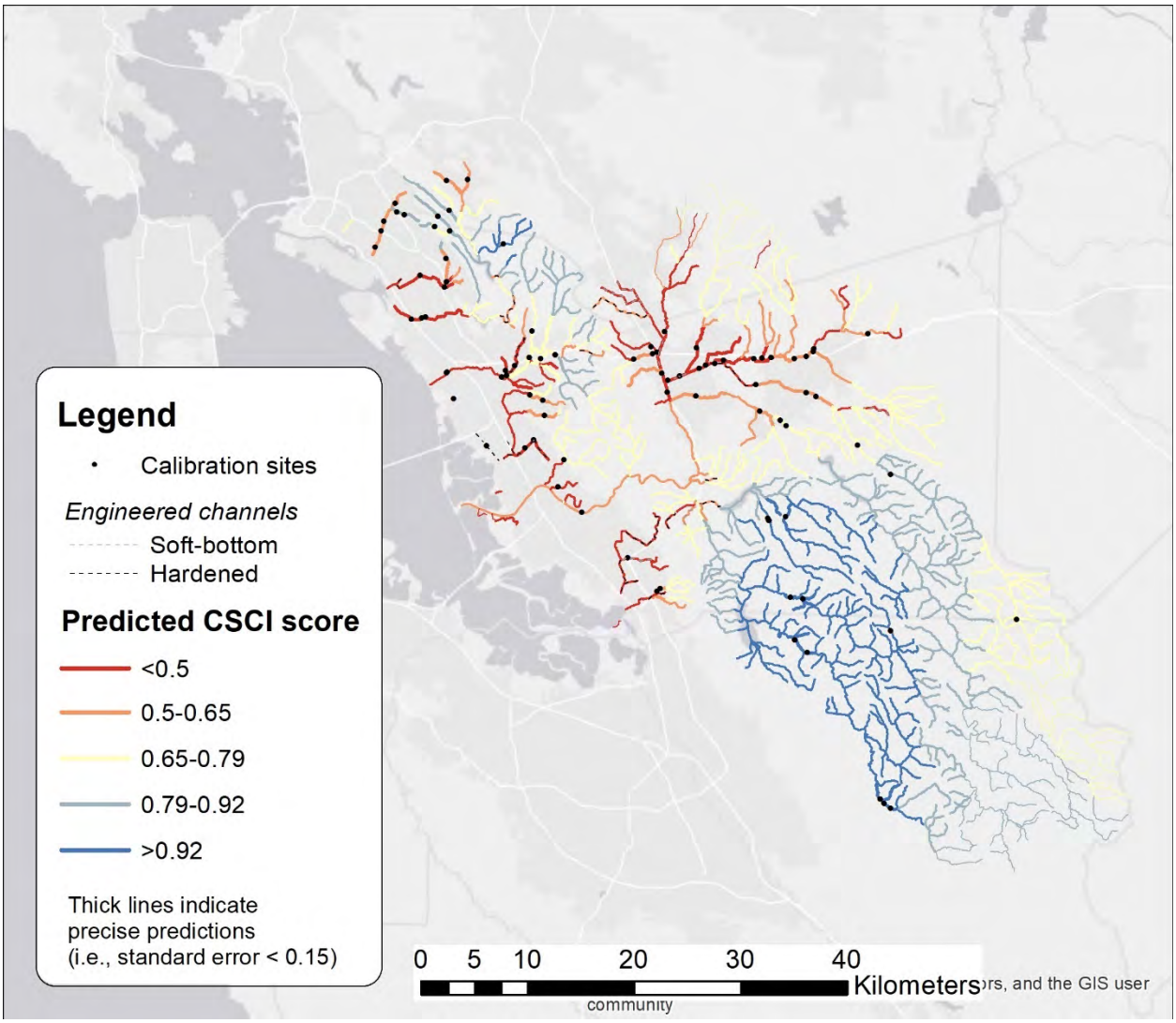


Figure 9. Predicted CSCI scores in Alameda County streams.

Nearly half of the watershed (i.e., 48% of stream-miles) were predicted to have CSCI scores below 0.79, comparable to Santa Clara county. Precision was better here than in any other case study: only 17% of stream-miles had poor precision (i.e., standard error >0.2), and 10% had good precision (i.e., standard error <0.15) (Figure 10). Like Ventura County, precise estimates were more frequent in engineered channels than in natural channels, although in Alameda County, soft-bottom channels were more precisely estimated than hardened channels (Table 12).

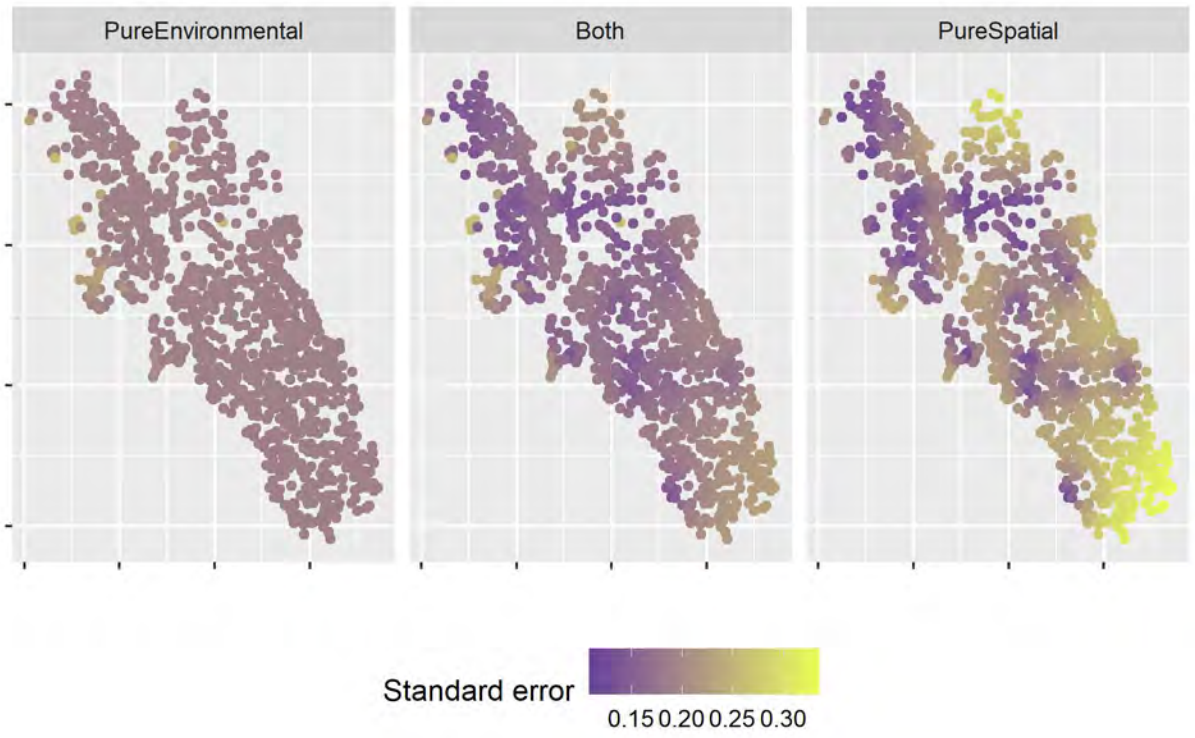


Figure 10. Prediction standard errors for the top models in Alameda County.

Table 12. Percent of stream-miles with high, medium and low precision (based on ranges of standard error shown in parentheses) in Alameda County.

Channel type	High (< 0.15)	Medium (0.15 to 0.20)	Low (> 0.20)
Natural	8	74	18
Engineered	29	69	2
Hardened	23	75	1
Soft	43	53	3

CITED LITERATURE

Beck, M.W., R.D. Mazor, S.J. Wisenbaker, J. Westfall, P.R. Ode, R. Hill, C. Loflen, M. Sutula, and E.D. Stein. 2019. Prioritizing management goals for stream biological integrity within the developed landscape context. Freshwater Science 2019 38:4, 883-898

Frieden, J.C., E.E. Peterson, J.A. Webb, and P. Negus. 2014. Improving the predictive power of spatial statistical models of stream macroinvertebrates using weighted autocovariance functions. Environmental Modelling and Software. 60: 320-330.

Hill, R.A., M.H. Weber, S.G. Leibowitz, A.R. Olsen, and D.J. Thornbrugh. 2016. The Stream-Catchment (StreamCat) dataset: A database of watershed metrics for the conternminous United States. Journal of the American Water Resources Association. 52(1): 120-128.

Mazor, R.D., P.R. Ode, A.C. Rehn, and E.D. Stein. 2017. Spatial statistical network models to estimate the spatial representativeness of bioassessment samples. SCCWRP Technical Report #979. Costa Mesa, CA. http://ftp.sccwrp.org/pub/download/DOCUMENTS/TechnicalReports/979_SpatialStatisticalNetworkModel2017.pdf

Mazor, R.D., A.C. Rehn, P.R. Ode, M. Engeln, K.C. Schiff, E.D. Stein, D.J. Gillett, D.B. Herbst, and C.P. Hawkins. 2016. Bioassessment in Complex Environments: Designing an Index for Consistent Meaning in Different Settings. Freshwater Science 35 (1): 249–71.

Nagel, D., E. Peterson, D. Isaak, J. Ver Hoef, and D. Horan. 2015. National Stream Internet Protocol and User Guide, version 3-22-2017. US Forest Service, Rocky Mountain Research Station. Boise, ID. https://www.fs.fed.us/rm/boise/AWAE/projects/NationalStreamInternet/downloads/NationalStreamInternetProtocolandUserGuide.pdf

Peterson, E.E. 2019. STARS: Spatial tools for the analysis of river systems version 2.0.7 – A tutorial. Queensland University of Technology. Brisbane, Australia. https://www.fs.fed.us/rm/boise/AWAE/projects/SSN_STARS/downloads/STARS/STARS_tutorial_2.0.7.pdf

Peterson, E.E. and J. M. Ver Hoef. 2014. STARS: An ArcGIS toolset used to calculate the spatial information needed to fit spatial statistical models to stream network data. Journal of Statistical Software, 56(2).

Ver Hoef, J.M., E.E. Peterson, D. Clifford, and R. Shah. 2014. SSN: An R Package for Spatial Statistical Modeling on Stream Networks. Journal of Statistical Software, 56(3), 1-43. http://www.jstatsoft.org/v56/i03/

APPENDIX: A GUIDE TO CREATING SPATIAL STATISTICAL NETWORK (SSN) MODELS FOR BIOASSESSMENT DATA IN CALIFORNIA WATERSHEDS

Who is this guide for?

This guide is written for technical staff at watershed management agencies working in California who are familiar with spatial statistical network models and want to develop them for watersheds in California. The guide provides links to resources, walks through major analytical steps, and provides support for interpreting results, all with a specific focus on evaluating bioassessment data in California. The guide is intended to supplement SSN training materials produced by Peterson (2019) and Ver Hoef et al. (2014).

Getting started

What software do you need?

To create SSNs, you need the following software:

1. ArcGIS version $\geq$ 10.6
2. Advanced license and the Spatial Analyst extension
3. STARS version $\geq$ 2.0.7 geoprocessing toolbox for ArcGIS (https://www.fs.fed.us/rm/boise/AWAE/projects/SpatialStreamNetworks.shtml)
4. Python version $\geq$ 2.7.14
5. PythonWin: must be downloaded and installed separately from Python. Go to this website: http://sourceforge.net/projects/pywin32/files/pywin32, click on Build 221, and download this file: pywin32-221.win32-py2.7.exe.
6. R version $\geq$ 3.5 (https://cran.r-project.org/)
7. The SSN package for R version $\geq$ 1.1.6 (https://cran.r-project.org/)

Data requirements

Certain datasets are required to generate the spatial data needed to fit a spatial stream-network model. As a convenience, several California-specific data sets have been bundled together in a package for download from SCCWRPs FTP site:

http://ftp.sccwrp.org/pub/download/PROJECTS/CASQA_SSN/CASQA_SSN_DataPackage.zip

Dataset	Format	Required	Where to get
Shapefile defining area of interest (e.g., watershed delineation, county boundary).	Polygon shapefile	Required	Provided by user
Topologically correct stream network hydrography (e.g., NSI).	Polyline shapefile	Required	Data package
Observed CSCI scores with latitude and longitude	CSV or excel spreadsheet with one row per site (no replicates)	Required	Data package (should be supplemented with additional data when possible)
Landcover data (e.g., StreamCat)	CSV spreadsheet with one row per COMID	Optional	Selected data from StreamCat appended to NSI in data package. Full StreamCat data are

			available here: https://www.epa.gov/national-aquatic-resource-surveys/streamcat
Channel engineering data	Polyline shapefile, with simple classes (e.g., natural vs. engineered, or natural vs. hard-bottom vs. soft-bottom) in attribute table	Optional	Provided by user

A shapefile representing the network: The National Stream Internet (NSI)

Although the National Hydrography Dataset Plus (NHD Plus, MacKay et al. 2012) represents nation-wide stream coverage with reasonably high spatial accuracy, it does not always reflect hydrologic connectivity with sufficient accuracy for SSN model development. For example, reaches may be isolated from each other, and braided sections may be represented as multiple stream-segments. Nagel et al. (2015) modified the NHD Plus to more accurately reflect these hydrologic relationships, as needed for spatial modeling.

The data package provides an excerpt of the NSI for the entire state of California, updated with environmental data from the StreamCat dataset (Hill et al. 2016, described below). In general, California watershed managers should not need to create or greatly modify the NSI for their watersheds. Exceptions may include watersheds that span international boundaries. The full NSI network for the United States can be accessed from https://www.fs.fed.us/rm/boise/AWAE/projects/NationalStreamInternet.html.

Observed data with location information

CSCI scores and location information are required to calibrate an SSN model. As a rule of thumb, we recommend having at least 30 locations sampled to attempt modeling, although smaller data sets may be useful as well.

The data package includes CSCI scores calculated at 3,254 sites throughout the state of California, dating from 2000 to 2016. Sources of data include bioassessment samples collected by the Surface Water Ambient Monitoring Program, the stream survey of the Stormwater Monitoring Coalition in southern California, and other samples found in the California Environmental Data Exchange Network (CEDEN).

California watershed managers may want to update these data as newer samples in the watershed of interest become available.

Landcover information

Although SSN models can be built without environmental covariates, models that include covariates tend to outperform purely spatial models (sometimes greatly so). Environmental covariates should be included only if there is a plausible relationship between the covariate and biological condition. For example, increased urban or agricultural land cover could degrade CSCI scores by altering watershed hydrology or introducing contaminants in a stream (Beck et al. 2019).

The StreamCat dataset (Hill et al. 2016) provides estimates of land use within the riparian zone (i.e., a 100-m buffer on each side of a stream segment), the local catchment (i.e., nearby landscape flowing directly into the immediate stream segment, excluding upstream segments), and the entire upstream watershed for each stream-segment in NHD Plus (the basis for the NSI). Many of the metrics in StreamCat were derived from the 2006 National Land Cover Database (Fry et al. 2011).

The NSI shapefile in the data package has been updated with selected metrics from StreamCat known to be useful for predicting CSCI scores. The full StreamCat data set can be accessed from https://www.epa.gov/national-aquatic-resource-surveys/streamcat. Note that any StreamCat metric expressed percentages (e.g., impervious surfaces, agricultural land use) must be converted to total area (e.g., km^2) in order to successfully run the STARS tools. Specifically, this should be done for catchment-scale metrics by multiplying them by the catchment area (in km^2), then dividing by 100. Metrics expressed as densities (e.g., road density, road crossing density) must be similarly treated by multiplying the catchment-scale metric by catchment area. As a convenience, the California data package includes these metrics in both percentage/density and raw forms.

Channel engineering information

Stormwater agencies may have shapefiles representing location and composition of engineered channels, typically for the purposes of tracking operations and maintenance of infrastructure. We provide instructions below on how to update the NSI with this information. Ideally, these data contain basic information about the composition of beds and banks, such that they can easily be grouped into a few classes (e.g., fully hardened vs soft bottom, with or without low-flow channels, etc.). The format and content of the data may vary widely among sources, so care should be taken when combining multiple shapefiles.

Creating an SSN

Reviewing your input data for adequacy

Before proceeding, check the following:

- Do you have the required software installed to run the STARS toolbox?
  - ArcGIS version 10.6 or higher
  - Python version 2.7.14 or higher
  - STARS toolbox 2.0.7 or higher
- Do you have the required data?
  - A shapefile delineating the area of interest
  - NSI hydrography
  - A spreadsheet with CSCI scores and coordinates
- Do you have the desired non-required data?
  - A spreadsheet with landcover data for each COMID
  - A shapefile with channel engineering information

Installing STARS package

The STARS geoprocessing toolbox is written in Python version 2.7.14 for ArcGIS version 10.6. The STARS toolbox (Figure 11) contains three toolsets: Pre-processing, Calculate, and Export.

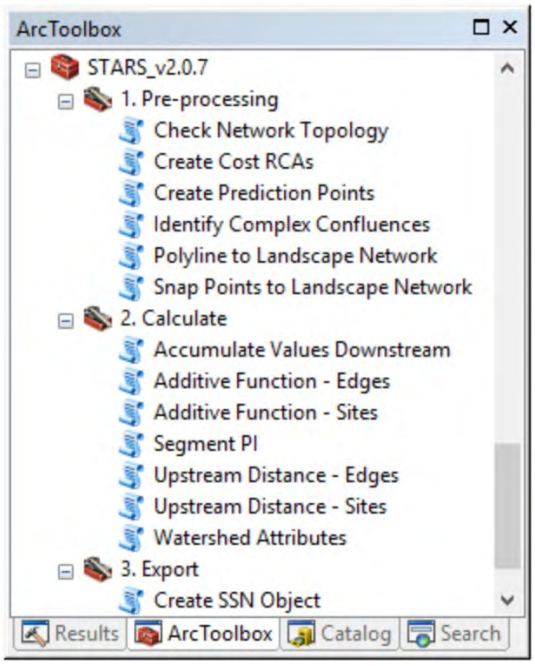


Figure 11. Contents of the STARS toolbox for ArcGIS. From Peterson (2019).

These tools are specifically designed to analyze, reformat, and export the spatial data as a .ssn ("dot s-s-n") object. The .ssn object can be directly imported to R statistical software (R Core Team 2019) using the SSN package (Ver Hoef et al. 2014), where spatial stream-network models can be fit to streams data.

1. Open ArcMap.
2. Add the STARS toolbox. In ArcToolbox, right click on 'ArcToolbox', and select 'Add Toolbox'. Navigate to the STARS toolbox and click OK. Then, right click on ArcToolbox, scroll down, select Save Settings, and click on To Default. Make sure that there are no spaces in the pathname where the STARS tools reside.
3. Change Environment Settings. Go to Menu, click on Geoprocessing, and select Environments. Expand M Values and set Output has M Values to Disabled. Repeat these steps for Z Values and click OK.
4. Overwrite outputs by default. In the menu, select Geoprocessing > Geoprocessing Options and check the box next to Overwrite the outputs of geoprocessing operations.

Data pre-processing

Open a new ArcMap session with a new to create a new LSN (doing so helps limit error). Also, make sure there are ***no spaces*** in the pathnames where the datasets reside (e.g., C:\MyData\gisdata is good; C:\My Data\gisdata is bad).

Prepare data inputs

1. Create a shapefile of observed CSCI scores.
    - If necessary, update the data in the data package with additional CSCI scores.
    - Import into ArcGIS as a points shapefile called "Observed_CSCI".
2. Customize the input shapefiles to the area of interest:
    - NSI. In general, we recommend using "intersect" rather than "clip", as clipping tends to introduce topological errors and artificial "outlets" in the stream network. See section on topological errors below for more details.
    - Observed data points and rename "Observed_CSCI_Clip".
3. If desired, update the NSI with additional environmental data at this point. The shapefile in the California data package has already been updated with basic landcover information, such as percent imperviousness.
4. Review shapefile for missing environmental data. Data gaps for a variable preclude its use in modeling. If data gaps can't be filled, it may be necessary to exclude stream segments (or environmental variables) from further analysis. Make sure that when you drop a stream segment, you avoid introducing topological errors (see Figure 12a below for examples of topological errors introduced by removing segments downstream of a confluence).

Update the NSI with channel engineering information

In general, stormwater infrastructure information is not registered to NHD Plus, meaning that some effort is required to update the NSI with this information. We describe a process for this update below.

1. If necessary, create a simple classification in the channel engineering shapefile. For example:
    - Natural vs. Engineered
    - Natural vs. Hard-bottom vs. Soft-bottom

   More complex classifications are not recommended.
2. If the channel engineering shapefile has COMIDs, use this first to update the NSI where COMIDs match.
    - Create a left-join based on COMID
    - Add a field called "Status" to the NSI. (Multiple fields may be added if multiple classifications are needed.)
    - Use Field Calculator to update this field with the appropriate field in the channel engineering shapefile.
    - Save the new shapefile.
3. If the channel engineering shapefile lacks COMIDs, use the Transfer Attributes tool found in the Editing Toolbox (see Figure 12).
    - Create a backup of the NSI feature.
    - In the Transfer Attributes tool, the Source Features is the channel engineering shapefile and the "Target features" is your copy of the clipped NSI layer. Set search distance to 50 m and leave all other fields blank. This will "transfer" the status from the channel engineering shapefile to the NSI based on a proximity of 50 m.

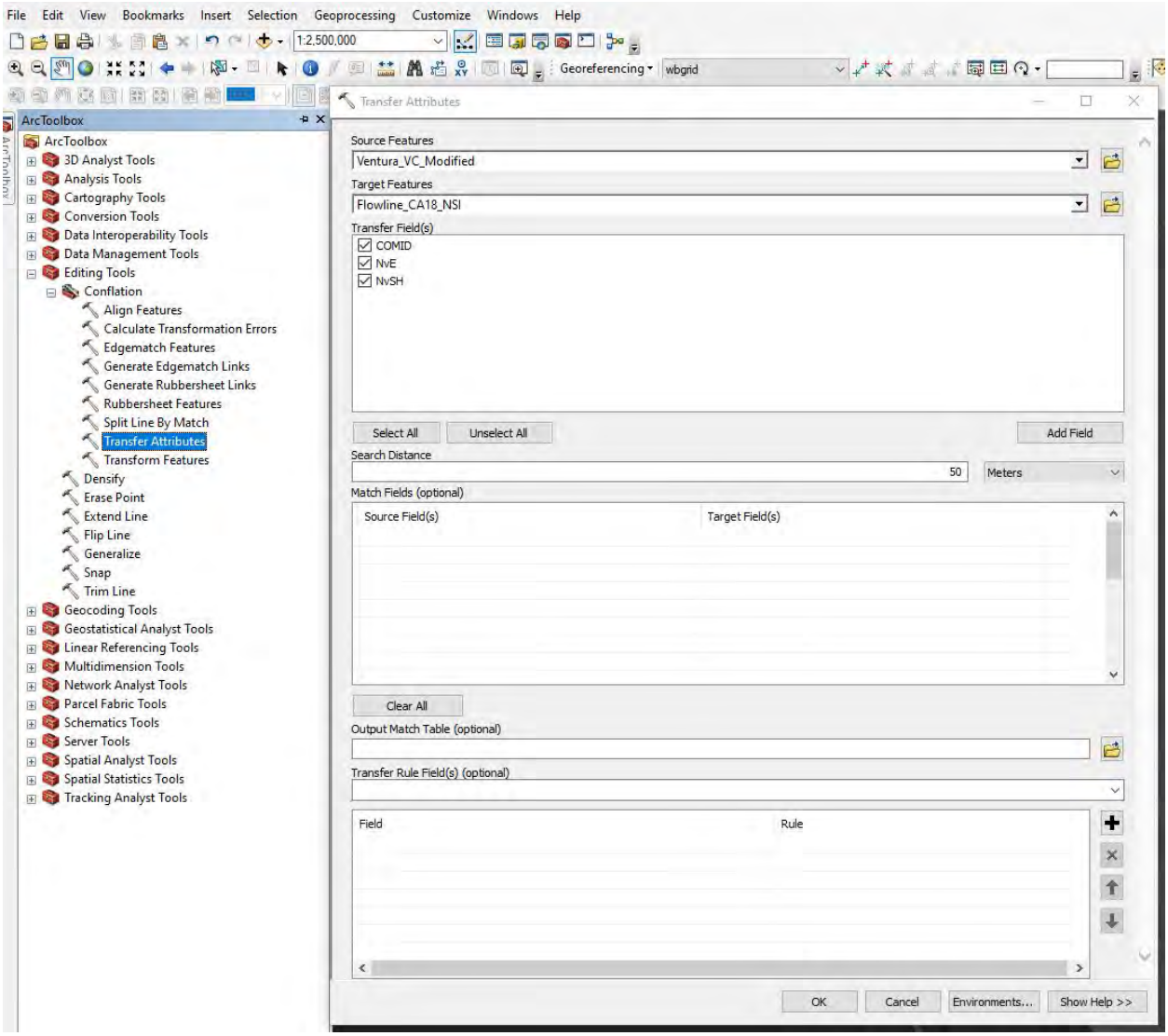


Figure 12. Example setup for the "Transfer Attributes" tool.

4. Several segments in the NSI will not have any assigned class. It may be sufficient to presume that these are natural channels. However, you can infer class from landcover or other information. For our case studies, we made the following assumptions:
    - Channel engineering shapefiles included all hard-bottom engineered channels. However, soft-bottom engineered channels could be missing.
    - These missing soft-bottom engineered channels were likely associated with a high degree of local imperviousness. Specifically, we assumed that stream segments with more than ~15% at the riparian-catchment scale (based on the Imp2011CRB field in the updated NSI layer that is included in the data package) is an engineered, soft-bottom channel.
5. Verify that there are no segments with missing data. Unless there is evidence to the contrary, assume that segments without channel engineering status represent natural channels.

After making these assignments, compare classifications to aerial imagery to assess the accuracy of the updated NSI. Make manual corrections as needed.

Creating a Landscape Network (LSN) in ArcGIS

A landscape Network (LSN) is a type of graph that is used to represent spatial context and relationships with additional geographic information (Theobald et al. 2006). An LSN is stored as an ESRI personal geodatabase and has four distinguishing features (Theobald et al. 2005):

- The LSN has the ability to store both the topology of a graph and the geometry of the nodes, reaches, and reach contributing areas (RCAs). Note that the terms “reach” and “segment” are both used to refer to a single line-segment in the polyline streams dataset (i.e., NSI).
- Nodes represent confluences, stream sources, or stream outlet points.
- Edges represent flow paths from node to node.
- RCAs represent the aerial extent that would theoretically contribute overland flow to a given edge in the absence of other hydrologic processes such as infiltration or evaporation.

Create the LSN

1. Load the required datasets into ArcGIS:
    a. The updated and clipped NSI shapefile
    b. The updated and clipped CSCI shapefile
2. Create a new folder to hold the LSN, with no other folders in this folder.
3. Open the “Polyline to Landscape Network” tool: STARS > Pre-processing >Polyline to Landscape Network
4. Set the pathnames and click OK (see Figure 12).
   Note: Do not use a shapefile named edges.shp. Doing so will lead to errors in the relationship tables of the LSN.

   Be sure to include the file extensions .shp and .mdb.

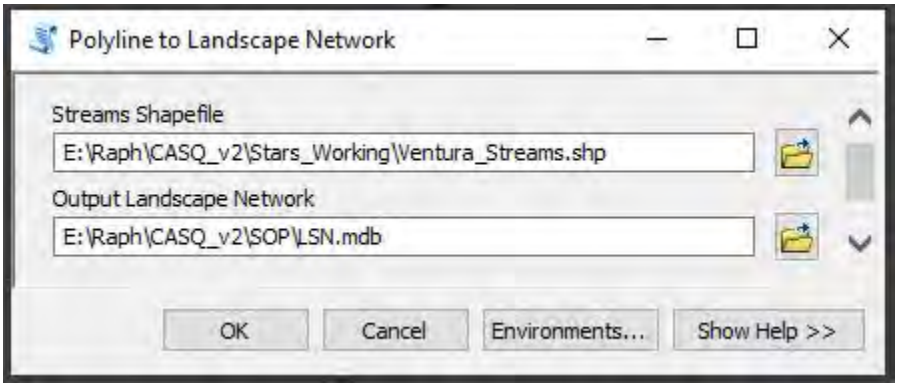


Figure 13. Example setup for the "Polylines to Landscape Network" tool.

5. Load the LSN into ArcMap.

   Note: An Unknown Spatial Reference warning box may appear indicating that the nodes feature class is missing spatial reference information. Click OK.

   If everything works correctly, the message FINISHED Polyline to Landscape Network Script will be shown in green. The tool produces a personal geodatabase with five components:

   - NODES (POINT FEATURE CLASS)
   - EDGES (POLYLINE FEATURE CLASS)
   - RELATIONSHIPS (TABLE)
   - NODERELATIONSHIPS (TABLE)
   - NODEXY (TABLE)

   The program fails to produce an LSN if any of these components are missing.

Review and correct topological errors

Although the NSI has been carefully edited to remove nearly all of the topological errors in the NHD Plus, a few may remain. These errors must be corrected to ensure that hydrologic distances and spatial relationships are calculated properly. There are two tools provided in the STARS toolset to help identify these errors:

- Check Network Topology
- Identify Complex Confluences

Topological errors are identified in the edges feature class, but they must be manually corrected in the streams.shp dataset that was used to build the LSN. The new LSN must be re-evaluated in an iterative process.

Correcting errors can be a difficult and time-consuming process. Despite its improvements over the NHD Plus, the NSI retains several errors, which may be important in certain regions. In addition, the act of subsetting the NSI for the region of interest may introduce new errors.

There are several types of errors, and the tools may flag errors that are in fact correct. For example, the tool may flag “outlets” (i.e., stream segments that flow into the ocean or outside the area of interest) that look like errors. Most of these errors can be safely ignored. However, one type of error requires close attention and must always be eliminated from an LSN: Converging streams. Convergence errors occur when two tributaries combine at a confluence, but the confluence has no outlet (Figure 12).

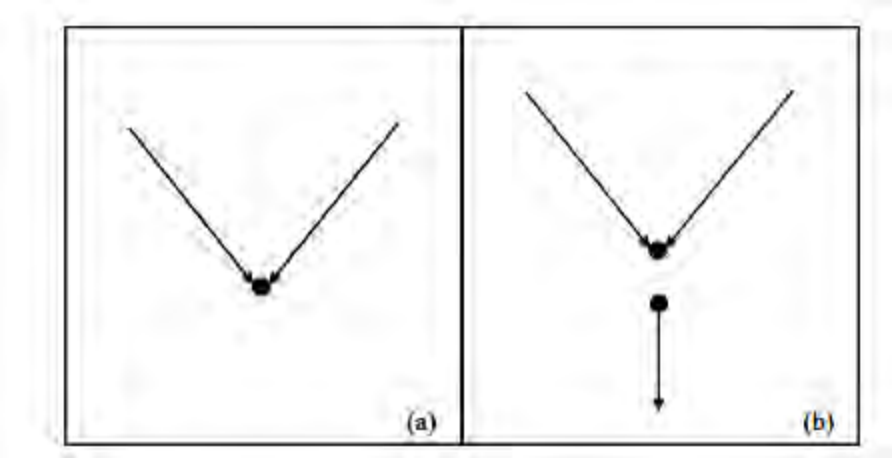


Figure 14. Examples of topological errors requiring connection. In both panels, two tributaries combine, but lack a downstream outlet. In panel A, the downstream segment is entirely missing, whereas in panel B, the downstream segment is disconnected from the node. Figure from Peterson (2019).

When modifying a shapefile that produces convergence errors, you have a few options:

- Re-subset the NSI, making sure to include the stream-segment downstream of the convergence. This is a good choice when the error was created by the subset process but won't help if the convergence error is intrinsic to the NSI.
- Delete the entire network upstream of the convergence. This is a good choice when the network is small and contains no data.
- Clip the NSI above the convergence, creating two new outlets
- Manually snap the upstream tributaries to the downstream segment.

Choosing an option depends on the amount of time available, your comfort editing shapefiles in GIS, and the importance of the affected area to management questions or analysis.

Once the errors are corrected in the shapefile, a new LSN must be built: It is not possible to 'fix' the LSN. Run the Check Network Topology and the Identify Complex Confluences tools on each new LSN to ensure that it is free of topological errors and convergent streams.

Note: The development of a topologically corrected stream network can be the most time-consuming aspect of the modelling process. You may need to try more than one approach to fixing topological errors. Some fixes may work one time then not work on a similar instance. As you work through this project, take note of which techniques resolved certain types of errors. Do your best to resolve all errors to the best of your ability but be aware of time vs. effect on final output. It's easy to get lost in a sea of topological errors.

Please see the STARS Tutorials for more information on identifying and resolving topological errors.

Generate prediction points

Prediction points are the locations where the SSN model will be used to estimate CSCI scores. In general, you want a high density of prediction points, with multiple points on every segment of the NSI. The latest version of the STARS toolkit includes the Create Prediction Points tool to facilitate the creation of prediction points.

If you already have a set of prediction points, move to the next section.

You need to use the original NSI shapefile with the .shp extension. Do not use the edges file in the LSN generated above.

1. Go to STARS v2.0.7 > Pre-processing > Create Prediction Points. This will open the Create Prediction Points window.

    a. Set the argument values (see Figure 15). Note that the interval is in map units and in this example the units are meters.

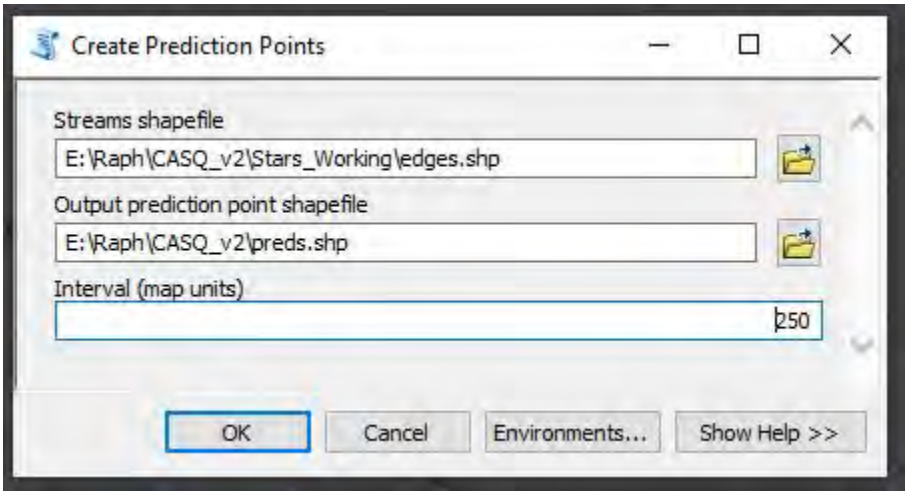


Figure 15. Example setup for the "Create Prediction Points" tool.

Incorporate points with observations into the LSN

1. Go to STARS > Pre-processing > Snap Points to Landscape Network. This will open the Snap Points to Landscape Network window.
2. Set the path names as needed. If the sites have already been edited and snapped to the appropriate edges, set the Search Radius = 1. Otherwise, examine the data and choose an appropriate search radius based on the maximum distance between sites and segments. We recommend keeping this radius below 10 m; larger distances may be ok but require careful review to ensure that sites are “snapped” correctly.

Note: Your path name may be different. Be sure that the file extensions .shp and .mdb are included

3. Click OK. The tool may take a few minutes to run.

If the tool runs successfully, the message Finished Snap Points to Landscape Network Edges will appear.

A new sites feature class will be written to the LSN geodatabase if the survey locations are successfully incorporated. The new sites attribute table will contain some new fields, two of which are the *ratio* and *rid* fields. The rid field indicates which edge the site has been snapped to. The ratio for each site provides the exact location along the edge.

4. Compare the total number of sites in the feature class to the total number of sites in the shapefile to ensure that all of the sites have been incorporated into the LSN.

Calculate watershed attributes

The watershed attributes assigned to the edges represent the watershed attribute for the downstream node of each edge. However, survey sites can fall anywhere along an edge. The Calculate > Watershed Attributes tool enables watershed attributes to be estimated for any site that has been incorporated into the LSN.

Streamcat data is presented in watershed area. We have provided 3 columns at the end that are site specific.

To convert attributes to a percentage simply divide it by the total watershed area at the site and then multiply it by 100 to get the percentage.

Go to STARS > Calculate > Watershed Attributes. This will open the Watershed Attributes window.

1. Set the argument values.
2. Watershed attributes may be calculated for multiple site feature classes simultaneously (i.e., observed and prediction sites).
3. The Edge Watershed Attribute Name must be an accumulated field found in the edges attribute table.
4. The Edge RCA Attribute Name should be the RCA field that was accumulated to produce the Edge Watershed Attribute Name field.

If the program finishes without errors, a green message will appear: Program finished successfully. A new field will be added to the sites attribute table that contains the watershed attribute for each site. If the tool is run separately for multiple sites feature classes (i.e., observed and then later prediction sites), ensure that the New Site Watershed Attribute Name is identical in all of the feature classes.

Calculate upstream distance

Calculate Upstream Distance – Edges

1. Double click on STARS > Calculate > Upstream Distance - Edges. This will open the Upstream Distance – Edges window.
2. Set the Edges Feature Class argument, select the Shape_Length attribute from the dropdown list, and click OK.

You should see a green message Program finished successfully. In addition, a new field should appear in the edges attribute table named *upDist*.

Calculate Upstream Distance – Sites

1. Double click on STARS > Calculate > Upstream Distance - Sites. This will open the Upstream Distance – Sites window.
2. Set the arguments and click OK.

If no errors occurred, a green message, Program finished successfully, will appear.

Note: The upDist attribute must be present in the edges attribute table before the Upstream Distance – Sites tool can run.

Calculate Segment Proportional Influence

1. Calculate the watershed area for each edge in the LSN using the STARS > Calculate > Accumulate Values Downstream tool (see section 12 for instructions). Set 'Field to Accumulate' to rcaAreaKm2 and name the new field 'h2oAreaKm2'.
2. Double click on the STARS > Calculate > Segment PI
3. Set the arguments (below) and click OK.

A green message will appear, Program finished successfully, if it runs without errors.

This tool adds a new field to the edges attribute table that contains the segment PI values. In this example, it is named areaPI.

If any PI values are greater than 1, then an error has occurred, and the segment PIs should be recalculated.

4. Check the areaPI field to ensure that values range between 0 and 1.
5. Open the edges attribute table, right click on the areaPI field, scroll down and select

Statistics. This will open the Statistics of edges window, which contains summary statistics for the areaPI field.

6. Examine the minimum and maximum values to ensure that they range between 0 and 1.

Calculate Additive Function

Calculate Additive Function – Edges

1. Go to STARS > Calculate > Additive Function - Edges. This will open the Calculate Additive Function - Edges tool.
2. Set the arguments and click OK.

If the script runs without errors, a green message will appear: Finished Get Additive Function Script.

Calculate Additive Function - Sites

1. Go to STARS > Calculate > Additive Function - Sites. This will open the Calculate Additive Function - Sites tool.
2. Set the arguments and click OK.

When the tool finishes running, a green message should appear: Finished Additive Function Script.

The Additive Function tools create new fields in both the sites and edges attribute table representing the AFV values. The AFV is a product of proportions (segment PI values) and so the AFV should always range between 0 and 1. Check to ensure that this is the case.

Create the SSN object

The purpose of the Create SSN Object tool is to reformat the LSN as a Spatial Stream Network (.ssn) object. The .ssn object represents the spatial data and the topology of the network in a format that can be easily accessed and efficiently stored and analyzed in R statistical software using the SSN package.

The .ssn object contains the spatial, attribute, and topological information of the LSN. It always contains at least two shapefiles edges and sites, as well as multiple text files containing the edge binary IDs.

To create an .ssn object:

1. Double click on STARS > Export > Create SSN Object. This will open the Create SSN Object tool.
2. Set the parameters and click OK.

The tool can be run with the "StationID" as the Site ID Field.

The Create SSN Object tool may take a while to run depending on the number of edges and sites in the LSN. When the tool has finished successfully, a green message, Successfully Finished Create SSN Object Script, will appear. An .ssn object will also be created in the same directory as the LSN used to create it.

Creating a model with your SSN in R

Once you've created an SSN object in ArcGIS, you can import it into R and use it to develop models that predict CSCI scores or other variables you included in your sites data set. Currently, the SSN package in R supports the development of generalized linear models (GLMs); more complex models (such as generalized additive models) are not currently supported.

Before you begin

The structure of SSN objects is very complex, so it is difficult to modify the data at this point. If you want to add new variables or experiment with transformations of the data, it is probably simpler to create a new SSN object by repeating the steps above than to modify one you've already created.

Software requirements

- R version 3.5 or later (https://cran.r-project.org/)
- SSN package for R, version 1.1.6 or later (https://cran.r-project.org/)
- R studio (recommended) (https://rstudio.com/)

Import the SSN object

To import an SSN object, you must use the importSSN() function and specify the path name. At the same time, you can import the prediction points as well with the predpts argument.

```
mySSN<-importSSN("P:/AbelSantana/Raph/LSN.ssn",
predpts="sitesPP")
```

This step may take a few minutes, particularly for large or complex SSNs.

Optional: Visualize spatial variability by creating torgegram

Spatial variability can be explored by examining pairs of sites with observed data, plotting variability on the y-axis and distance separating the sites on the x-axis. These types of graphs are called variograms, and when they are applied to SSNs, they are called torgegrams (Figure 13). Examining a variogram or torgegram can give a sense of whether spatial persistence is high (i.e., sites are similar, even when far apart) is low (i.e., sites differ greatly, even when close together).

Use the `Torgegram()` function to create a torgegram:

```
myTorg<-Torgegram(object=mySSN, ResponseName="CSCI")
```

where object is the SSN object, and ResponseName is the name of the variable you want to predict.

You can visualize the torgegram with the `plot()` function:

```
plot(myTorg)
```

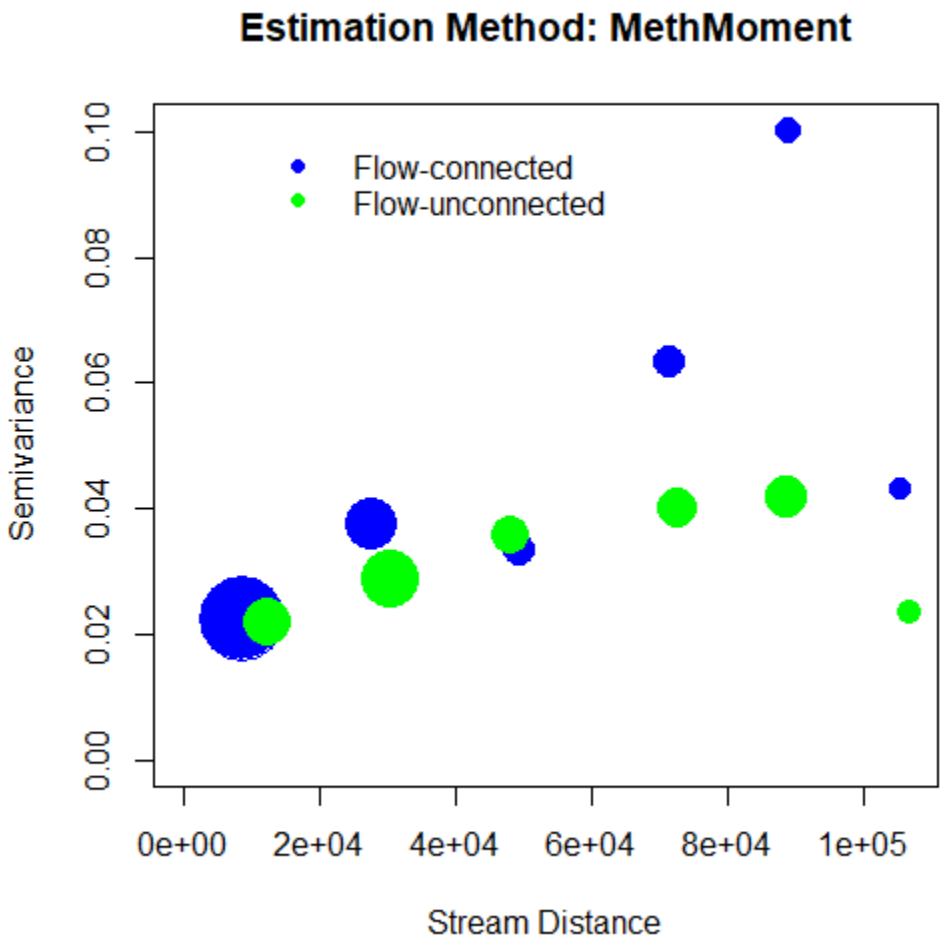


Figure 16. Torgegram showing how semivariance of CSCI scores relate to stream distance in the Santa Clara watershed. Dots show the mean standard deviations of CSCI scores between pairs of points separated by distances shown on the x-axis. The size of the dot is proportional to the number of pairs of sites used in the calculation. Green dots show pairs of sites not connected by flow (e.g., two sites on adjacent tributaries above a confluence), and blue dots show flow-connected pairs (e.g., sites that are up- or downstream of each other).

The semivariance (i.e., the standard deviation between pairs of sites) is plotted separately for flow-connected sites (blue dots) and for flow-unconnected sites (green dots); the size of the dot corresponds to the numbers of pairs used to calculate the standard deviation.

Calibrating the model

There are many ways to calibrate a model and determine the best combination of predictor variables for optimizing model performance. Whatever approach you use, it is generally easier to first select the best non-spatial model, then improve it by adding spatial components. The SSN package has a built-in function called `InfoCritCompare()` to facilitate comparisons among several models based on a variety of factors, such as the Akaike Information Criterion (AIC). We

will walk through examples of how to use this function to select the best non-spatial model, and then compare it to equivalent spatial models.

In this example, we consider three environmental factors to predict CSCI scores:

- PctImp20_1: The % imperviousness in the watershed from 2011 NLCD data.
- NvE: The status of the channel (natural vs. engineered)
- NvSH: The status of the channel (natural vs. soft-bottom vs. hard-bottom)

Calibrate a non-spatial model

We use the `glmssn()` function to create non-spatial models based on each of these predictors, one at a time:

```
glmssn.imp<-glmssn(CSCI~PctImp20_1, mySSN, CorModels = NULL,
EstMeth ="REML")

glmssn.nve<-glmssn(CSCI~NvE, mySSN, CorModels = NULL, EstMeth
="REML")

glmssn.nvsh<-glmssn(CSCI~NvSH, mySSN, CorModels = NULL, EstMeth
="REML")
```

Model formulas are specified using standard R notation, with the dependent variable (CSCI) to the left of the tilde, and the independent variables to the right. By specifying `CorModels = NULL`, the resulting model has no spatial component. By specifying `EstMeth ="REML"`, all estimates are based on restricted maximum likelihood.

We can also use standard formulas to specify two-term models to account for both land cover and channel engineering.

```
glmssn.nve_imp<-glmssn(CSCI~NvE + PctImp20_1, mySSN, CorModels =
NULL, EstMeth ="REML")

glmssn.nvsh_imp<-glmssn(CSCI~NvSH + PctImp20_1, mySSN, CorModels
= NULL, EstMeth ="REML")
```

If we believe that land cover and channel engineering have a combined impact, we can also specify interaction terms:

```
glmssn.nve_i_imp<-glmssn(CSCI~NvE * PctImp20_1, mySSN, CorModels
= NULL, EstMeth ="REML")

glmssn.nvsh_i_imp<-glmssn(CSCI~NvSH * PctImp20_1, mySSN,
CorModels = NULL, EstMeth ="REML")
```

To compare all these models, first combine them into a list:

```
list.mods_nonspatial<-list(glmssn.imp,

            glmssn.nve,
```

```
                glmssn.nvsh,
                glmssn.nve_imp,
                glmssn.nvsh_imp,
                glmssn.nve_i_imp,
                glmssn.nvsh_i_imp)
```

Then use the `InfoCritCompare()` function:

```
infocrit_nonspatial <- InfoCritCompare(list.mods_nonspatial)
```

```
> infocrit_nonspatial
                   formula EstMethod Variance_Components  neg2LogL       AIC          bias
1        CSCI ~ PctImp20_1      REML              Nugget -152.2253 -150.2253 -3.422169e-04
2              CSCI ~ NvE      REML              Nugget -142.6791 -140.6791 -6.568667e-15
3             CSCI ~ NvSH      REML              Nugget -155.3590 -153.3590  3.459278e-15
4  CSCI ~ NvE + PctImp20_1      REML              Nugget -188.5840 -186.5840 -3.125566e-04
5 CSCI ~ NvSH + PctImp20_1      REML              Nugget -195.8483 -193.8483 -2.771469e-04
6  CSCI ~ NvE * PctImp20_1      REML              Nugget -212.6763 -210.6763 -2.853227e-04
7 CSCI ~ NvSH * PctImp20_1      REML              Nugget -206.7404 -204.7404 -9.464088e-03
       std.bias     RMSPE       RAV  std.MSPE    cov.80    cov.90    cov.95
1 -5.828792e-04 0.1925380 0.1922659 1.0013893 0.8044077 0.9063361 0.9614325
2 -1.482996e-14 0.1962416 0.1963062 0.9996728 0.7961433 0.9090909 0.9559229
3  7.881775e-15 0.1919232 0.1919694 0.9997020 0.8099174 0.8980716 0.9559229
4 -5.478179e-04 0.1820097 0.1818647 1.0007835 0.7988981 0.9228650 0.9531680
5 -4.900370e-04 0.1791946 0.1791133 1.0003984 0.7988981 0.9201102 0.9531680
6 -5.102794e-04 0.1751214 0.1750937 1.0001549 0.8016529 0.9146006 0.9586777
7 -7.633197e-03 0.2438505 0.1941608 1.0073089 0.7988981 0.9008264 0.9586777
```

Several useful model evaluation metrics are included in this object, a few of which are described here.

Akaike's Information Criterion (AIC) is a widely used measure of quality for statistical model that quantifies the amount of information provided by a model, penalized by its complexity (i.e., the number of predictors). The lower the number indicates the more informative model. In the example shown above, the model in the 6th row (i.e., the one based on an interaction between percent imperviousness and a two-state classification of channel engineering) is best.

Bias is a measure of how different predicted CSCI scores are from observed scores, averaged across observations. Ideally, bias is close to zero. Bias is very low in all examples shown above.

Root Mean Square Prediction Error (RMSPE) is a measure of error in predictions. Lower values are better. Again, the model in the 6th row is best, although the 5th row is close behind.

cov.80, *cov.90*, and *cov.95* are measures of how frequently observed values were within the 80th, 90th, or 95th prediction intervals, respectively. Ideally, these proportions should be close to 0.8, 0.9, or 0.95, respectively.

When selecting a final model, consider a number of selection criteria, as well as other factors, such as the interpretability or ease of calculation of the selected predictors.

Based on these criteria, we identify the best non-spatial model as the one based on an interaction between percent imperviousness and a two-state classification of channel engineering:

```
CSCI ~ NvE * PctImp20_1
```

Create spatial versions of the selected non-spatial model

The `glmssn()` function can now be used to calibrate models that include a spatial component. There are two major factors to consider when calibrating an SSN model.

1. What spatial components do you want to use?
   There are three spatial components that can be included in an SSN model: Euclidean, tail-up, and tail-down. Euclidean components describe overland "as the crow flies" distances between sites. Tail-up components describe flow-connected distances from downstream sites to upstream sites within a network. Tail-down components describe flow-connected and flow-unconnected distances among sites within a network. Because CSCI scores are based on benthic macroinvertebrates, which can disperse upstream, downstream, and overland, it makes sense to consider all three components as a starting point.
2. What function should be used to approximate these components?
   The SSN package includes many options for approximating these components, but in general, we recommend using one of two simpler functions: spherical functions (which are appropriate when spatial persistence declines gradually with distance), and exponential functions (which are appropriate when spatial persistence declines abruptly with distance).

As a general recommendation, we suggest including all three spatial components with exponential functions as a starting point and exploring other combinations as-needed.

Calibrate models using the same `glmssn()` function, but this time, specify spatial components and functions:

```
glmssn.nve_i_imp_etu.etd.eeu<-
     glmssn(CSCI~NvE * PctImp20_1, mySSN,
          CorModels = c("Exponential.tailup", "Exponential.taildown",
               "Exponential.Euclid"),
          EstMeth ="REML",
          addfunccol="afvArea")
```

In this example, the three spatial components are specified with the `CorModels` argument. Whenever a tail-up component is included, you also need to use the `addfuncol` argument to specify which variable to use to define spatial weights; typically, these weights are based on area, meaning that the variable is called `afvArea`, if the LSN is created as described above.

Use the `InfoCritCompare()` function to compare the spatial model and non-spatial model. In this example, we create a model that includes the same predictors we selected before, plus Euclidean and a tail-down spatial components. Then, we compare the two models:

```
> glmssn.nve_i_imp_etd.eeu<-glmssn(CSCI~NvE * PctImp20_1, mySSN, CorModels = c("E
xponential.taildown","Exponential.Euclid"), EstMeth ="REML", addfunccol="afvArea"
)
> InfoCritCompare(list(glmssn.nve_i_imp,glmssn.nve_i_imp_etd.eeu))
                  formula EstMethod
1 CSCI ~ NvE * PctImp20_1      REML
2 CSCI ~ NvE * PctImp20_1      REML
                                Variance_Components  neg2LogL       AIC
1                                           Nugget -212.6763 -210.6763
2 Exponential.taildown + Exponential.Euclid + Nugget -376.3207 -366.3207
           bias      std.bias     RMSPE       RAV std.MSPE    cov.80    cov.90
1 -0.0002853227 -0.0005102794 0.1751214 0.1750937 1.000155 0.8016529 0.9146006
2  0.0003789994  0.0007441929 0.1308863 0.1301758 1.001969 0.8099174 0.9090909
     cov.95
1 0.9586777
2 0.9559229
```

The spatial model is considerably improved over the non-spatial version by most measures, particularly its greatly reduced AIC and RMSPE.

Save the best model in your working directory:

```
best.mod<-glmssn.nve_i_imp_etd.eeu
save(best.mod, file="best.model.Rdata")
```

Troubleshooting

If you were unable to remove or address all topological errors in the stream network, the resulting SSN object may contain asymmetrical distance matrices, which may prevent you from creating models with a tail-up component. In these cases, you will receive an error as follows:

```
> glmssn.nve_i_imp_etu<-glmssn(CSCI~NvE * PctImp20_1, mySSN, CorModels
= c("Exponential.tailup"), EstMeth ="REML", addfunccol="afvArea")
Error in fn(par, ...) : covariance matrix is not positive definite
```

You can either go back to the stream network and correct the topological errors (perhaps even by excluding the problematic segments from the analysis), or proceed without a tail-up spatial component (e.g., simply include Euclidean and tail-down components). The latter solution may be the most expedient way forward.

Getting useful info out of your model

Estimating variance components

Analyzing variance components can help determine the relative usefulness in predicting CSCI scores from spatial vs. non-spatial information. The varcomp() function is an easy way to access this information:

```
> varcomp(best.mod)
                VarComp  Proportion
1    Covariates (R-sq) 0.009826274
2 Exponential.taildown 0.128202814
3   Exponential.Euclid 0.670901807
4               Nugget 0.191069104
```

Environmental factors are represented in the first rows, labeled “Covariates”. In this example, information about landscape and channel structure provided only a small marginal benefit to predicting CSCI scores, compared to information about CSCI scores at nearby sites. The small variance component associated with covariates is typical of spatial models in environmental sciences.

Making Predictions

The `predict()` function can estimate CSCI scores (along with their standard errors):

```
my.preds_spatial<-predict(best.mod, predpointsID="sitesPP")
```

To extract these predictions for plotting in ArcGIS or other programs, use the following code:

```
my.preds_spatial.df<-
     cbind(getSSNdata.frame(my.preds_spatial,"sitesPP"),
     as.data.frame(mySSN@predpoints@SSNPoints[[1]]@point.coords))
```

You can just as easily generate predictions from the non-spatial models:

```
my.preds_nonspatial<-predict(best.nonspatial.mod,
     predpointsID="sitesPP")

my.preds_nonspatial.df<-
     cbind(getSSNdata.frame(my.preds_nonspatial,"sitesPP"),
     as.data.frame(mySSN@predpoints@SSNPoints[[1]]@point.coords))
```

At this point, it may be useful to classify prediction-points based on the predicted CSCI score (e.g., greater than 0.79 vs. lower) or standard error (e.g., high vs low precision):

```
my.preds_spatial.df$CSCI_class<-
     ifelse(my.preds_spatial.df$CSCI<0.79, "poor score","good
     score")

my.preds_spatial.df$Precision<-
     ifelse(my.preds_spatial.df$CSCI_se<0.15, "high
     precision","low precision")
```

You can export this dataframe with the `read.csv()` function and import it into ArcGIS to make a map. The coordinates are in the original projection of the spatial features used to create the LSN (e.g., UTM11, a typical projection for California).

Creating a map

Open a new session of ArcGIS, and load these files (Figure 14):

- The final modified NSI file or the edges feature in the LSN database.
- The prediction points, with predictions and standard errors as a points shapefile

Figure 17. Example ArcGIS workspace showing NSI flowlines and points with CSCI predictions for the Santa Clara watershed.

Use the Data Management Tools > Feature Class > Integrate tool.

Add the two shapefiles as inputs, and set the XY tolerance to 1 meter (Figure 15):

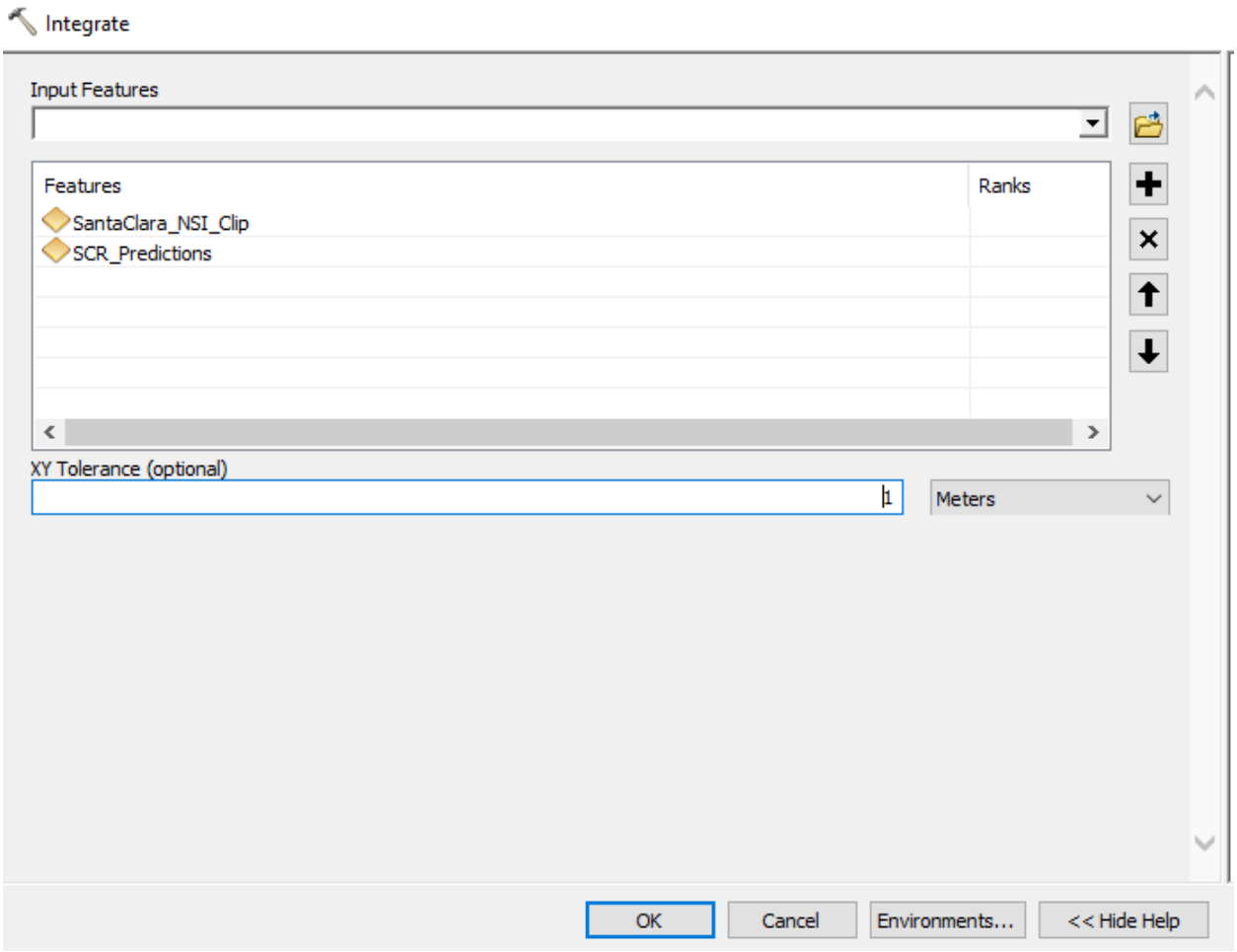


Figure 18. Set-up for the "Integrate" tool.

Use the Data Management Tools > Features > Split Line at Point tool.

Enter the NSI layer for the Input Features, and the prediction points for the Point Features. For Search Radius, enter 1 meter (Figure 16).

Because you have split each line up, it may be good to create a new field indicating the line length using the “Calculate Geometry” tool.

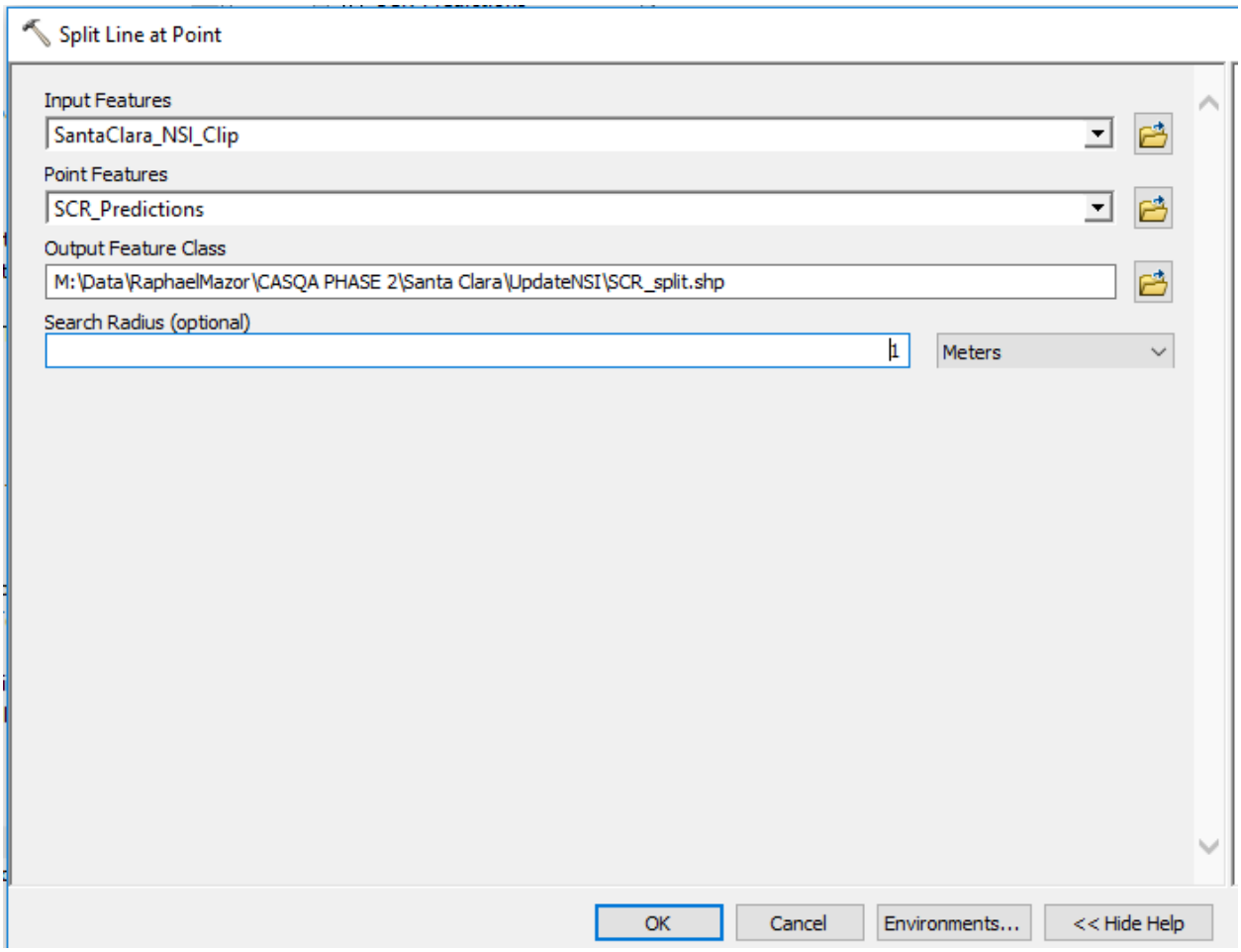


Figure 19. Set-up for the "Split Line at Point" tool.

Use the Analysis Tools > Overlay > Spatial Join tool.

Enter the newly split line layer for the Target Features, and the prediction points for the Join Features. Join Operation should be JOIN_ONE_TO_ONE, and “Keep All Target Features” should be checked. You can select the specific fields you want, or include all of them. Set Match Option to “INTERSECT” and “Search Radius” to 1 m—do not leave this blank! (Figure 17)

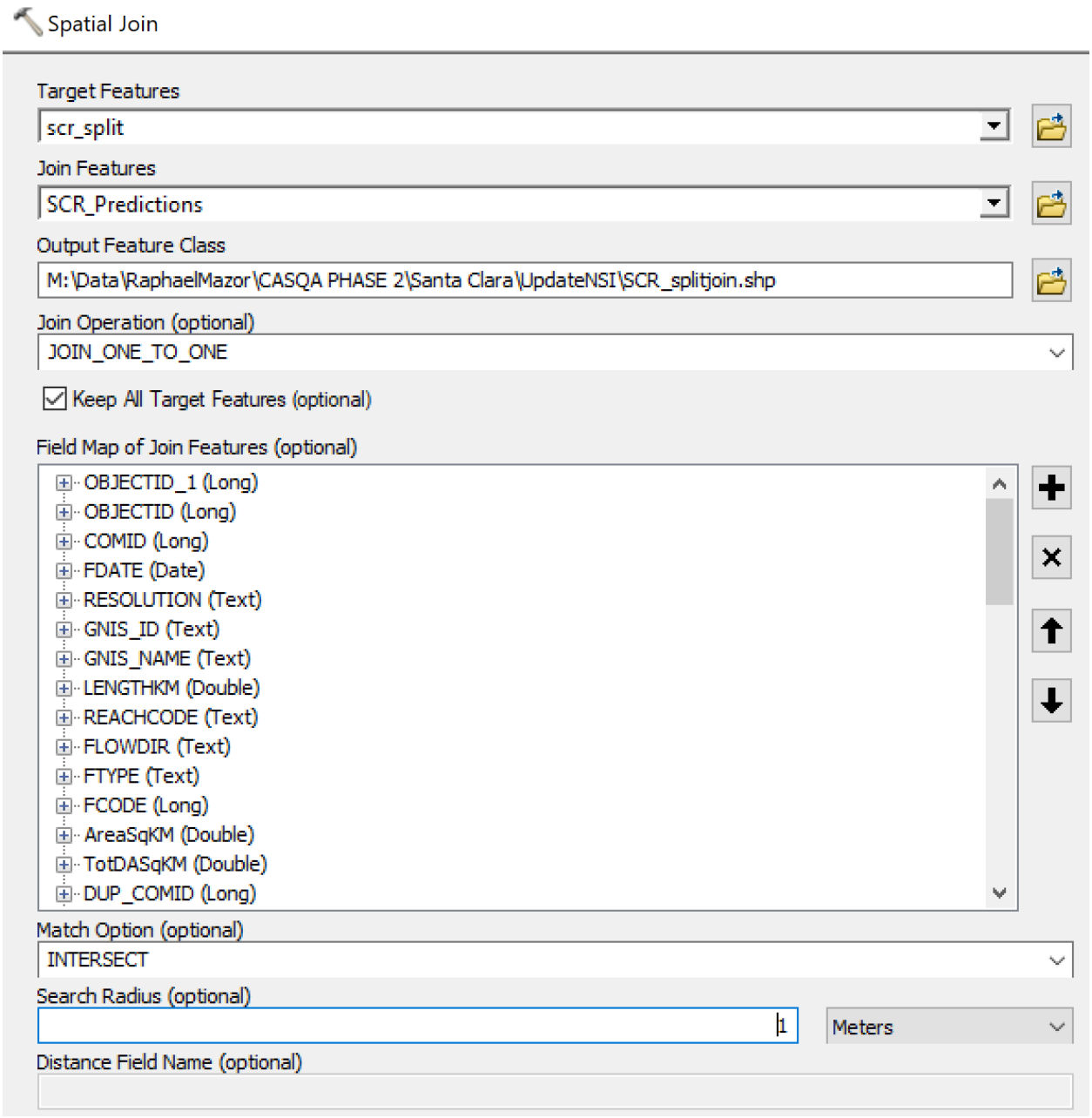


Figure 20. Set-up for the "Spatial Join" tool.

The resulting shapefile should have CSCI predictions and standard errors associated with each segment of the NSI (Figure 18).

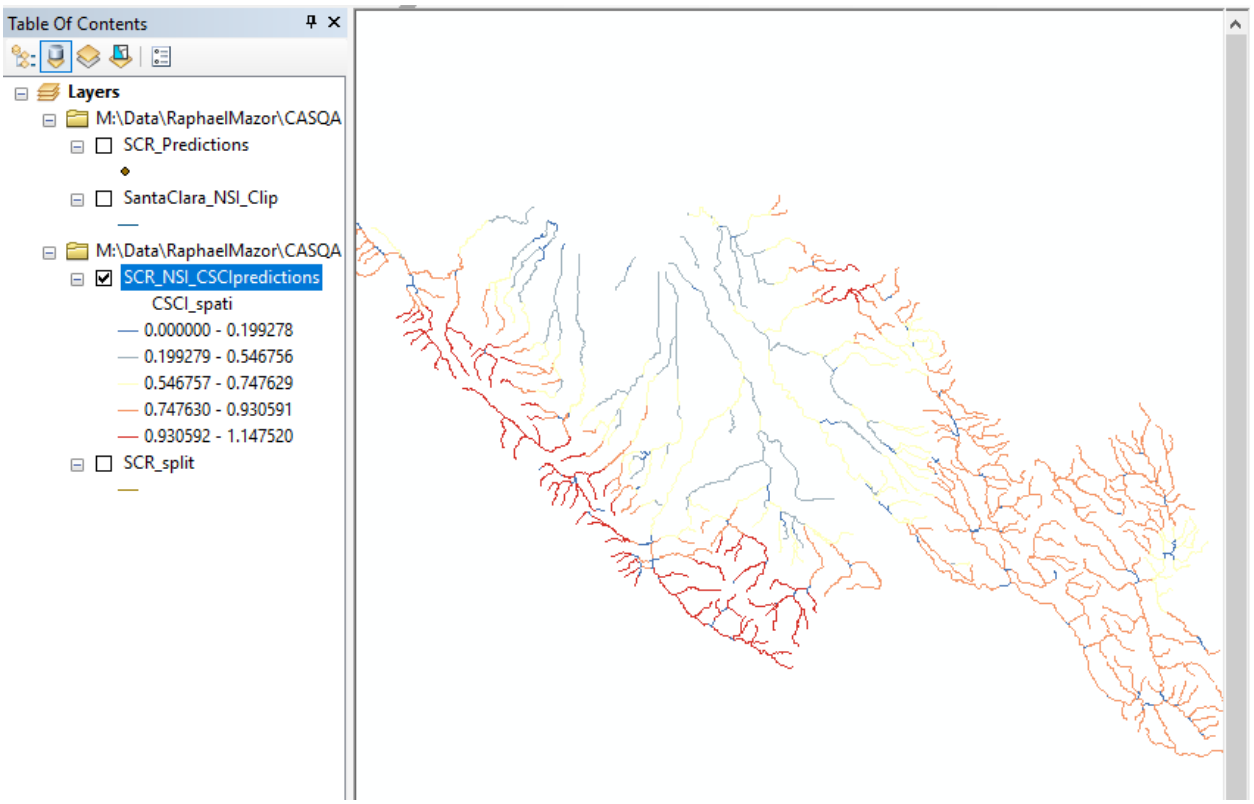


Figure 21. Example output showing flowlines updated with predicted CSCI scores.