CITY TREES: SEATTLE

Condition Prediction Project

Problem Statement

The worlds biggest and best cities are most often known for their buildings and infrastructure. When you think of New York you may think about the Empire State Building, Dubai probably makes you think of the Burj Khalifa. An important, and sometimes overlooked, part of our cities are the trees.

Trees play important roles in urban areas by improving air quality, regulating temperature, limiting stormwater damage, reducing noise pollution, and promoting biodiversity. They can also add beauty and provide space for community gatherings in areas often surrounded by seas of concrete.

Maintaining what researchers call our "urban tree canopy" is very important for the health of our urban communities. As we are facing climate change, the question becomes: what environmental factors impact the health of our urban tree canopy? And can we predict what that health will be in years to come?

This report pulls data from multiple sources in attempt to answer those very questions starting with one city: Seattle, Washington.

Data Wrangling

Trees Data

Our primary dataset is a subset of more than 5 million tree records published by university students here: https://datadryad.org/stash/dataset/doi:10.5061/dryad.2jm63xsrf

The data were split by city, allowing us to get down to more than 160,000 tree records, with 28 columns. Of those 28 columns, 13 were completely blank. This left me with latitude and longitude to pinpoint the locations, the planted and most recently observed dates, the diameter, as well ask some other categorical data to describe the tree and its location.

To supplement the data, I calculated an "age at observation" field by subtracting the planted date from the observation date and added a target feature called "condition_index" which converts the categorical condition values to a numerical format for predicting. These changes trimmed the features down to 10.

Climate Data

To supplement the dataset of trees, I pulled in 15-year climate normals from NOAA publicly available data located here: https://www.ncei.noaa.gov/data/normals-annualseasonal/2006-2020/

This data, broken out by weather station, gives important context on temperature and precipitation accross the United States. The first challenge is that it downloads for all available stations in a .csv file for each. I used a glob function to iterate through each .csv and load into a single dataframe. I then filtered down that dataframe to the stations in King County, WA (which Seattle is in), and trimmed fields down to temperature and precipitation averages, maxes, and mins.

The next challenge was the data only had climiate normals for three stations in the Seattle metro area, which didn't give me much specificity for a wide ranging area. to supplement this further I pulled 2022 precipitation and snowfall data from this source:
https://merbgai.cocorahs.org/ViewData/TotalPrecipSummary.aspx

This source gave me 28 stations to map into my trees data, but also brought its own challenges as I moved into exploratory analysis

Exploratory Data Analysis

Climate Data Cleanup

Our second climate source came from CoCoRaHS, which standard for the Community Collaborative Rain, Hail, & Snow Network. It is a crowd-sourced website for reporting daily data, with varying levels of completeness. This wasn't my idea source but created a great opportunity to learn how to work with incomplete data.

I broke this down into a few steps, first calculating the number of reports, applying an adjustment factor for the possibility of non-reported days having rain or not, and filling missing values with daily averages from close weather stations.

I then used the lat-long of our stations and trees to map each of my tree records to its nearest record station.

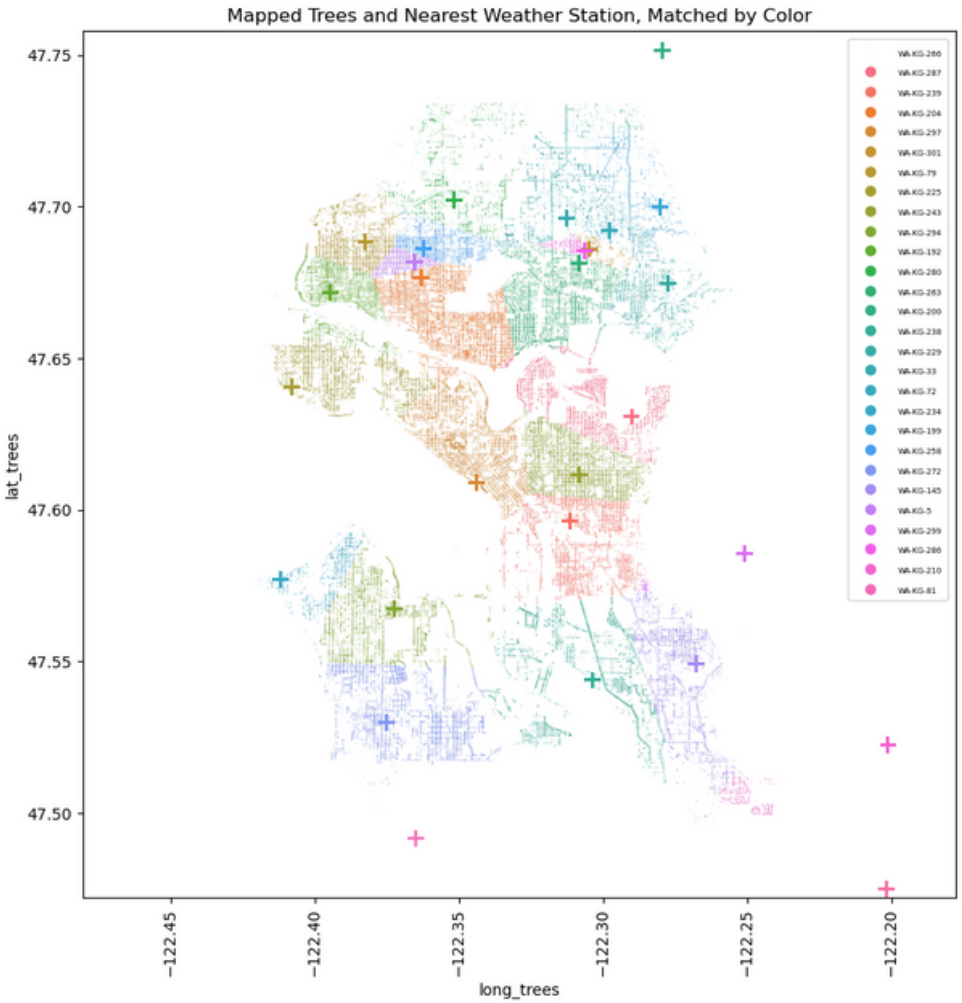


Figure 1: Visualization of trees and their nearest weather station

Exploring Features

One of the key features available in the primary tree dataset was diameter.

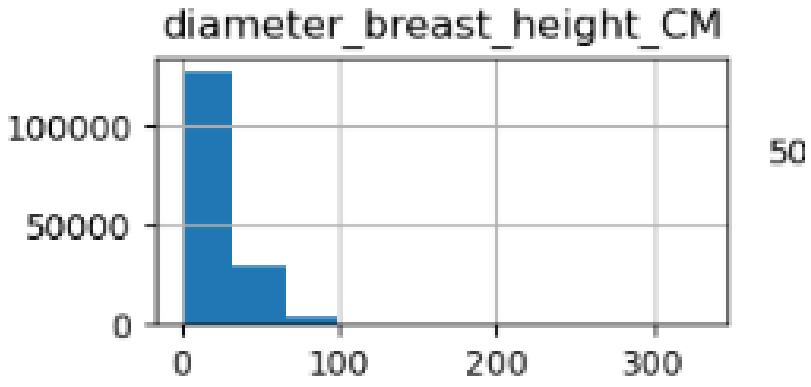


Figure 2: Distribution of tree diameter values

Having such a wide tail on our diameter distribution had me worried about potential outliers, so I took a deeper look with a different visualization of the aggregate and some individual species.

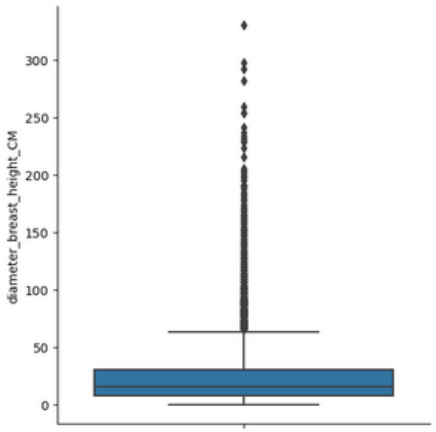


Figure 3: Boxplot of all records.

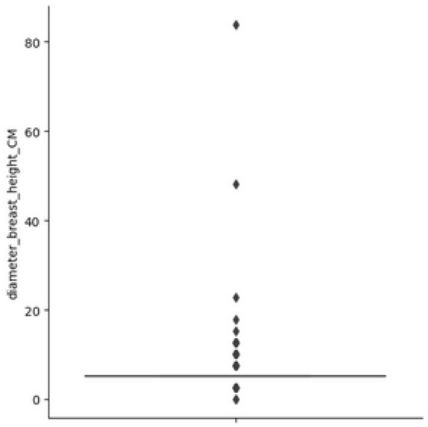


Figure 4: Only Eddie's Dogwood

The aggregate boxplot gave me a more powerful representation of our tailed data. But it was getting into the detail by species that I realized many records had identical diameters. It's possible the large scale of the data required some generalization/binning over precise measurements.

I like to use a 1.5X IQR definition for flagging outliers, but this binning made many of the IQRs equal to 0. To address this, I used the 1.5X IQR flag if for IQRs > 0 and > 3 standard deviations from the mean for IQRs = 0. I then dropped what this function flagged as an outlier, dropping 4,076 records (~2.5% of records).

Correaltion Matrix

To close out my EDA, I plotted a correlation matrix. Unfortunately I did not find an exciting correlation. Instead, just the covariance from some of the climate data I pulled into the analysis.

I dropped the duplicative values, leaving me with a final tally of 158,004 rows and 11 columns.

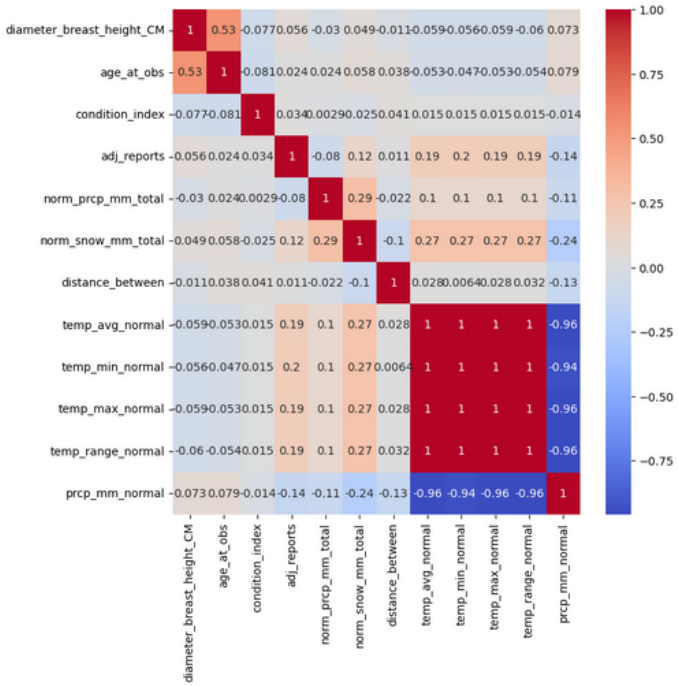


Figure 5: Correlation Matrix

Preprocessing & Training

Preprocessing

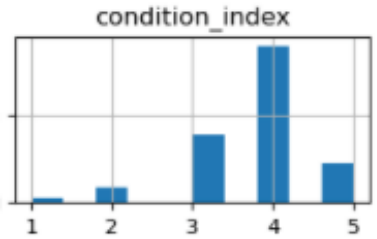
Having this many records proved to be slow when fitting models, so I sampled down to 10,000 records to start. After some trial and error I settled to a few consistent pre-processing steps:

- SimpleImputer using median values for numerical features
- SimpleImputer using 'missing' for categoricalfeatures
- StandardScaler on my tree age and climate fields (more normally distributed)
- PowerTransformer on my other numerical fields (more tailed distribution)
- OneHotEncoder on my categorical values, ignoring unknowns

The categorical encoding became my biggest challenge to transform only my two categorical features, but still scale to new data. I ended up using a custom process using OneHotEncoder, but saving the results to a new dataframe. It works well for the project but will need some tweaking before scaling more.

First Model

As a test, I ran my first model using Logistic Regression. It returned a training accuracy of 0.58, not too bad. But... my data is very imbalanced, so it did not predict any of my lesser classes.



	Precision	Recall	F1-Score	Support
1.0	0.00	0.00	0.00	120
2.0	0.00	0.00	0.00	421
3.0	0.48	0.19	0.27	1,921
4.0	0.59	0.94	0.73	4,410
5.0	0.56	0.15	0.24	1,128
Accuracy			0.58	8,000

Figure 6: Distribution of tree conditions in raw data
Figure 7: Classification report of first model

Balancing Data and Building Model

To address the imbalance in my classes, I used a Synthetic Minority Oversampling Technique (SMOTE) function called SMOTENC which takes in categorical features. This gave me an equal numbers of records in each class for training purposes.

I then ran this balanced data kFold cross-validation on multiple model types to see from where the best results could come and found random forest to be the most promising.

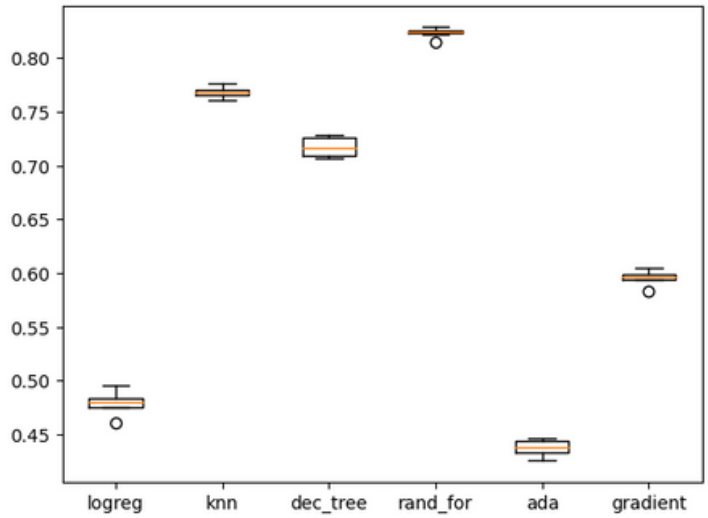


Figure 8: Boxplots of cross-validation results from different types of models.

Model Tuning

Focused on a random forest model, I first used RandomizedSerachCV to get a better idea of how to tune my hyperparamters. The returned best parameters gave a training accuracy and F1 of 100%, and I found max_depth to be the paramater that had the largest wiggle room.

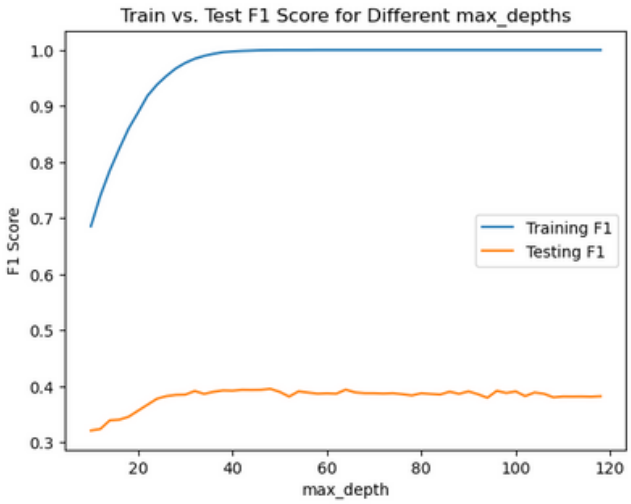


Figure 9: Training & testing f1 score at different max depths.

In an effort to generalize better to new data, I trimmed the tree down to a max of 40 depth and ran GridSearchCV. This returned a best model using:

- n_estimators = 1,000
- max_depth = 40
- min_samples_split = 2
- min_samples_leaf = 1
- max_features = 'sqrt'

As a final check I graphed a learning curve to see if I should use a larger sample.

Figure 10: Learning curve showing more data would benefit us

Final Model

With more data, my final model returned a test set accuracy of 0.59 and macro F1 of 0.42. It's likely even more data would help. But to not burn out my computer, I stuck with 20,000.

	Precision	Recall	F1-Score	Support
1.0	0.18	0.28	0.22	58
2.0	0.18	0.17	0.17	224
3.0	0.46	0.41	0.43	958
4.0	0.72	0.72	0.72	2,194
5.0	0.55	0.61	0.57	566
Accuracy			0.59	4,000

Figure 11: Classification report for final model on test data

Modeling

By pulling in .shp map file of the greaterSeattle area, I started by visualizing our actual and predicted tree conditions on the map.

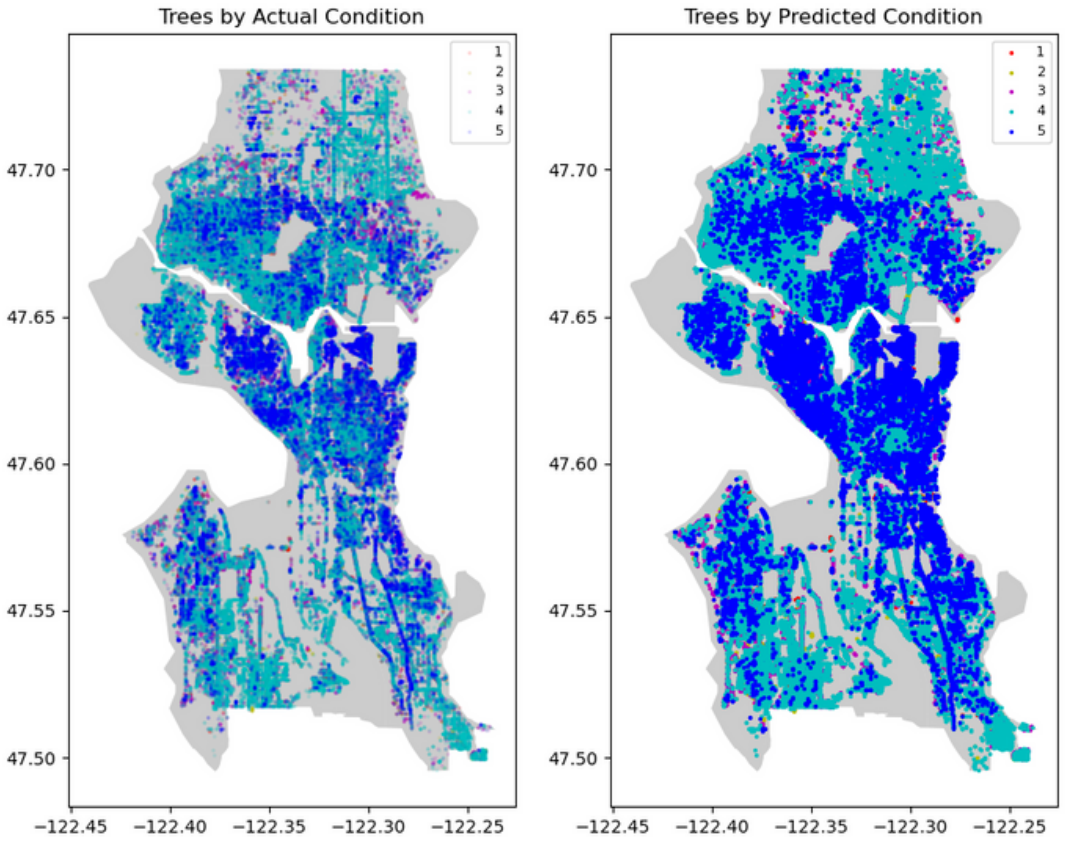


Figure 12: Comparing actual vs. predicted condition on the map.

You can see the heaver concentration of blue where the model predicts more 4 and 5 conditions. But this is only where the fun begins. Now that we have our model in place, we can use it to predict into the future.

Model 1: At Risk Projection

On the 1-5 scale, the values correspond to:

1. Dead/dying
2. Poor
3. Fair
4. Good
5. Excellent

At this scale, I flagged 1s and 2s as "at risk." I then add 25 years to each tree's age, and leaving all else equal, predicted at-risk trees in 2048.

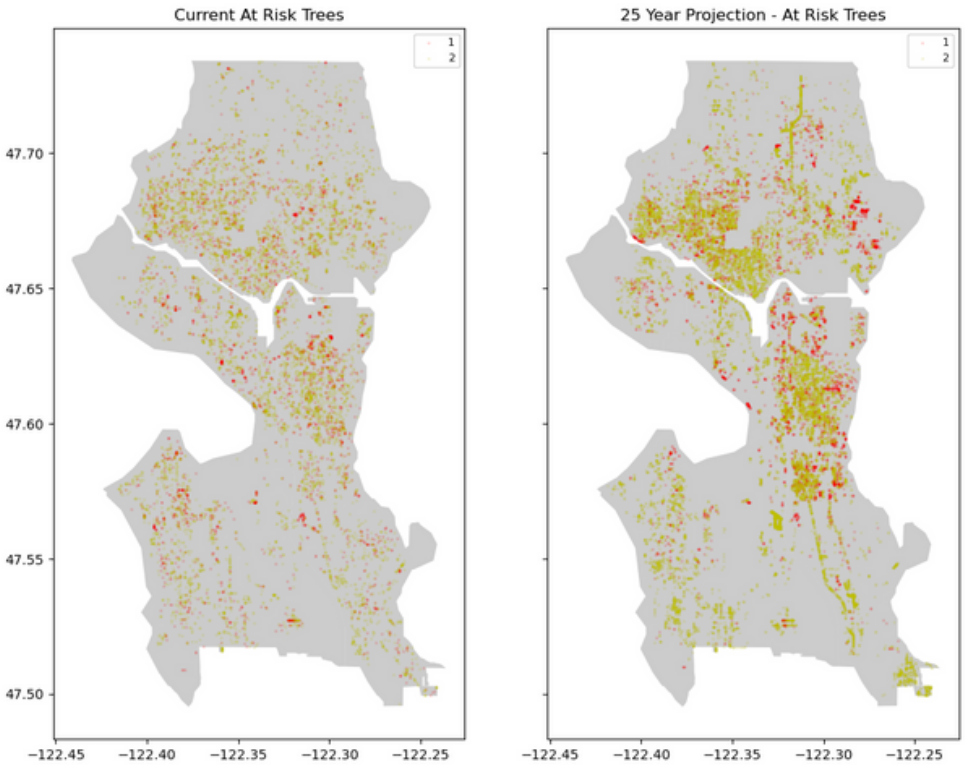


Figure 13: The 25-year projection shows a higher concentration of at-risk trees.

Assuming no changes in climate, our at-risk % goes from 7.01% to 13.73%. Of course we know climates are changing, so let's apply that fact to our model.

Model 2: Model Changing Climate Impacts

I made the following climate changes (on top of the 25-year fast forward):

- Average temperature: +5%
- Total current year rainfall: +5%
- Total current year snowfall: +2%
- Average long-term average rainfall: +8%

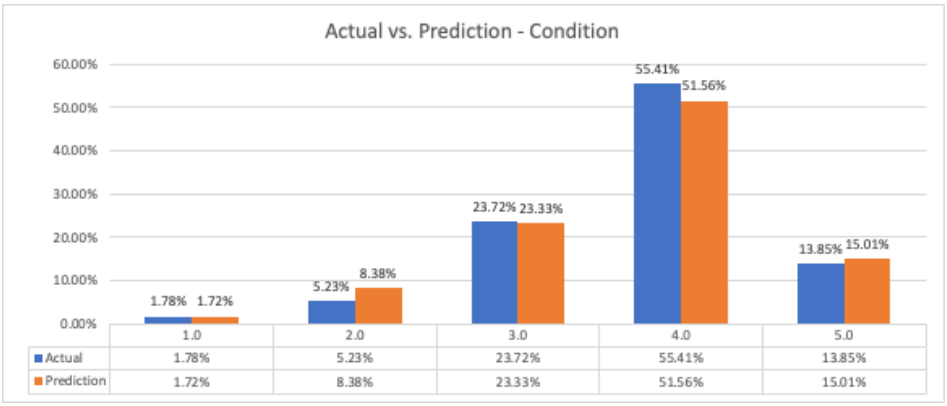


Figure 14: Predicted increases in the Poor and Excellent categories

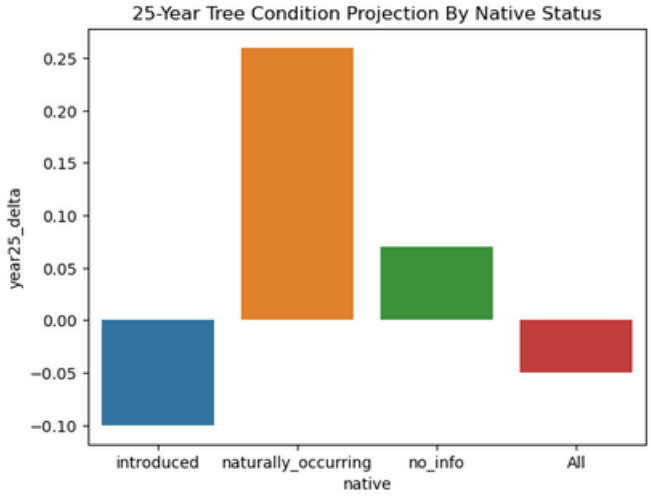
If these general climate prediction hold true, we could actually see some species of trees thriving more than they are now. Others, however will shift to the at-risk categories. This leads to the question of what types of trees are most at risk. We can answer this question by using the same model, but breaking out the details by type of tree.

Model 3: What Types of Trees are Most At-Risk

I started by subtracting the current actual quality index from the predicted values to get a 25-year delta. I then grouped the data by the species name and whether it is native or not. And found the average 25-year delta for each species. There were of course some outliers for species with only a few records, so I limited the query to only include species with at least 50 records.

The biggest drop is predicted for the Cascade Snow Cherry, with 58 trees dropping an average of 2.5 (4.0 down to 1.0 bin). The largest group in our top 10 predicted condition drops is the Forest Pansy Redbud, with 408 trees dropping an average of 1.0 (4.0 down to 3.0 bin).

These are just a few examples, but in further digging you can see the species that are naturally occurring are predicted to fair much better than those that have been introduced (the majority).



Native Category	Count	Actual Avg	Predicted Avg	Delta
Introduced	116,127	3.79	3.68	(0.11)
Naturally Occuring	9,261	3.66	3.92	0.26
No Info	32,616	3.61	3.68	0.07

Figure 15: Introduced species are the bulk of Seattle's tree cover, but predict to be in worse condition in 25 years

Future Steps

This model stands to improve greatly from more detailed climate data, which was hard to come beyond the weather station detail.

Some other geographic feature engineering was beyond my scope, but it could be useful to calculate clusters of trees using the latitude and longitude and how the species diversity within the trees and the geographic features around those clusters impact their health.

This project took a few turns from when I first had the idea. It tought me the value of building a more robust problem statement before getting excited about data. But it also gave me a lot of challenges and changes to learn new skills. With some further development it could help local officials intervene on at-risk trees before its too late to keep our urban tree canopies healthy.