Predicting the health of trees in New York City

Silvia Maione

Springboard Data Science Career Track

Outline

Dataset

Project goals
Questions

Data
wrangling

EDA

Model

Conclusions

Dataset

- https://data.cityofnewyork.us/Environment/2015-Street-Tree-Census-Tree-Data/uvpi-gqnh

tree_...	stum...	curb_...	status	health	spc_l...	spc_c...	stew...	guards	side...	user_...	probl...	root_...
3	0	OnCurb	Alive	Fair	Acer rubr...	red maple	None	None	NoDamage	TreesCou...	None	No
21	0	OnCurb	Alive	Fair	Quercus ...	pin oak	None	None	Damage	TreesCou...	Stones	Yes
3	0	OnCurb	Alive	Good	Gleditsia ...	honeyloc...	1or2	None	Damage	Volunteer	None	No
10	0	OnCurb	Alive	Good	Gleditsia ...	honeyloc...	None	None	Damage	Volunteer	Stones	Yes
21	0	OnCurb	Alive	Good	Tilia amer...	American...	None	None	Damage	Volunteer	Stones	Yes
11	0	OnCurb	Alive	Good	Gleditsia ...	honeyloc...	1or2	Helpful	NoDamage	Volunteer	None	No
11	0	OnCurb	Alive	Good	Gleditsia ...	honeyloc...	1or2	Helpful	NoDamage	Volunteer	None	No
9	0	OnCurb	Alive	Good	Tilia amer...	American...	None	None	NoDamage	Volunteer	MetalGra...	No
6	0	OnCurb	Alive	Good	Gleditsia ...	honeyloc...	None	None	NoDamage	TreesCou...	None	No
21	0	OffsetFro...	Alive	Fair	Platanus ...	London p...	None	None	NoDamage	TreesCou...	None	No
11	0	OnCurb	Alive	Good	Platanus ...	London p...	None	None	NoDamage	Volunteer	None	No
8	0	OnCurb	Alive	Poor	Platanus ...	London p...	None	None	NoDamage	Volunteer	None	No
13	0	OnCurb	Alive	Fair	Platanus ...	London p...	None	None	NoDamage	TreesCou...	Stones	Yes
22	0	OnCurb	Alive	Good	Platanus ...	London p...	3or4	Harmful	NoDamage	Volunteer	RootOther	No

Dataset Description

- 2015 New York City trees survey
- Almost 700k observations
- 45 features/columns, mostly categorical (species, health, branch/trunk/roots problems, etc.), plus tree’s diameter
- Geographical information also available (borough, latitude, longitude)

Project Questions and Goals

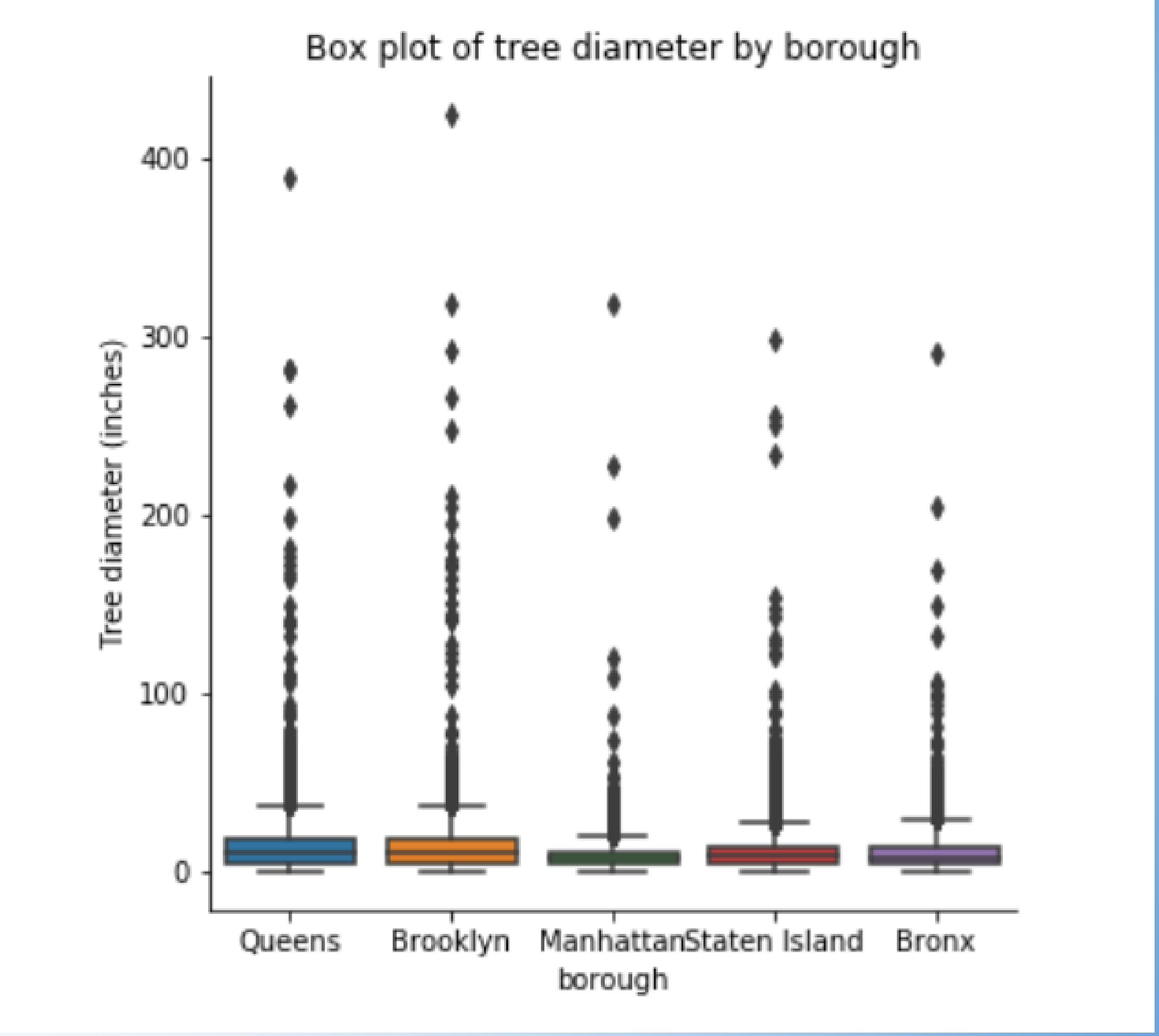
- How many trees?
- What’s the average diameter?
- What’s the most common species?
- Where are most of the trees?
- Is it possible to predict a tree’s health (good, fair or poor) knowing the location, the diameter, and the presence of any problems?

Data Wrangling

Missing entries are removed (not many)

Some entries have a very large diameter (not many)!

Categorical variables need to be encoded



EDA – Species

There are 132 unique species in New York City!

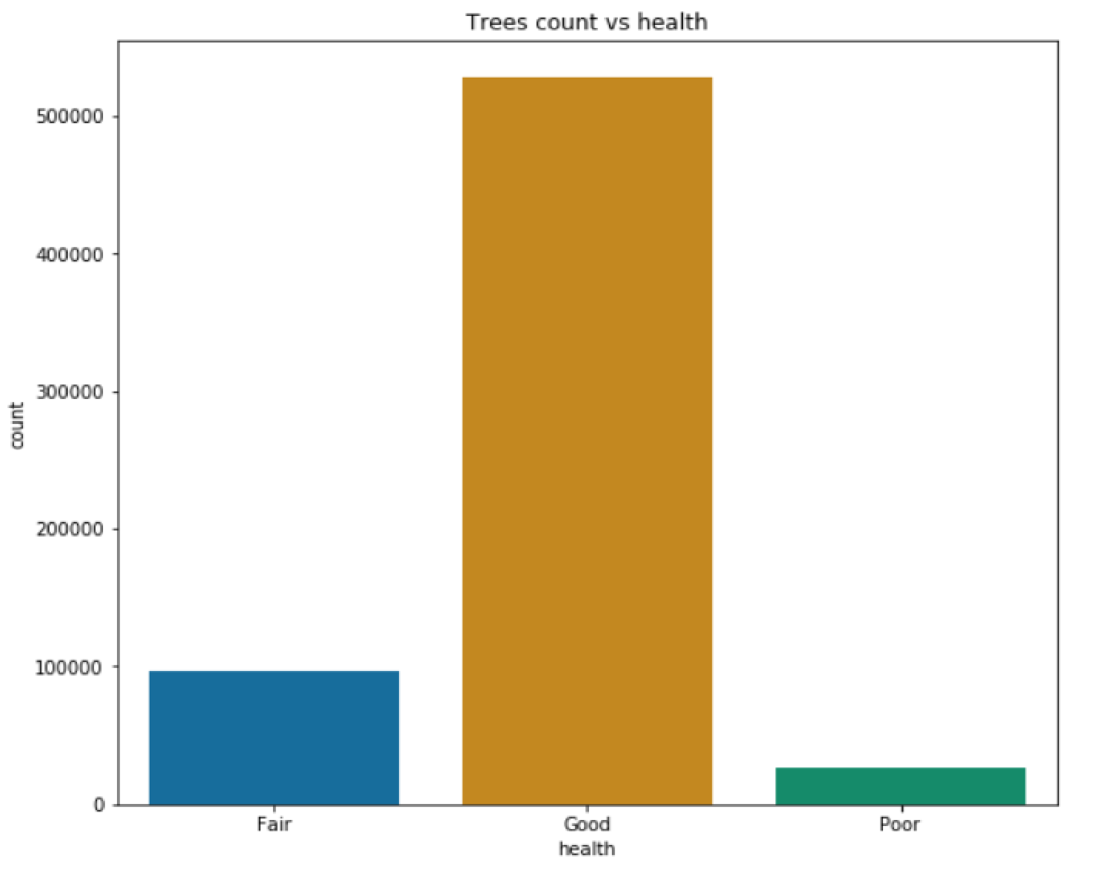
The most common species is the London planetree (87000 individuals)

The least common species is the Virginia pine (only 10!)

EDA – Health

The dataset is skewed, most trees are classified as healthy (85%)

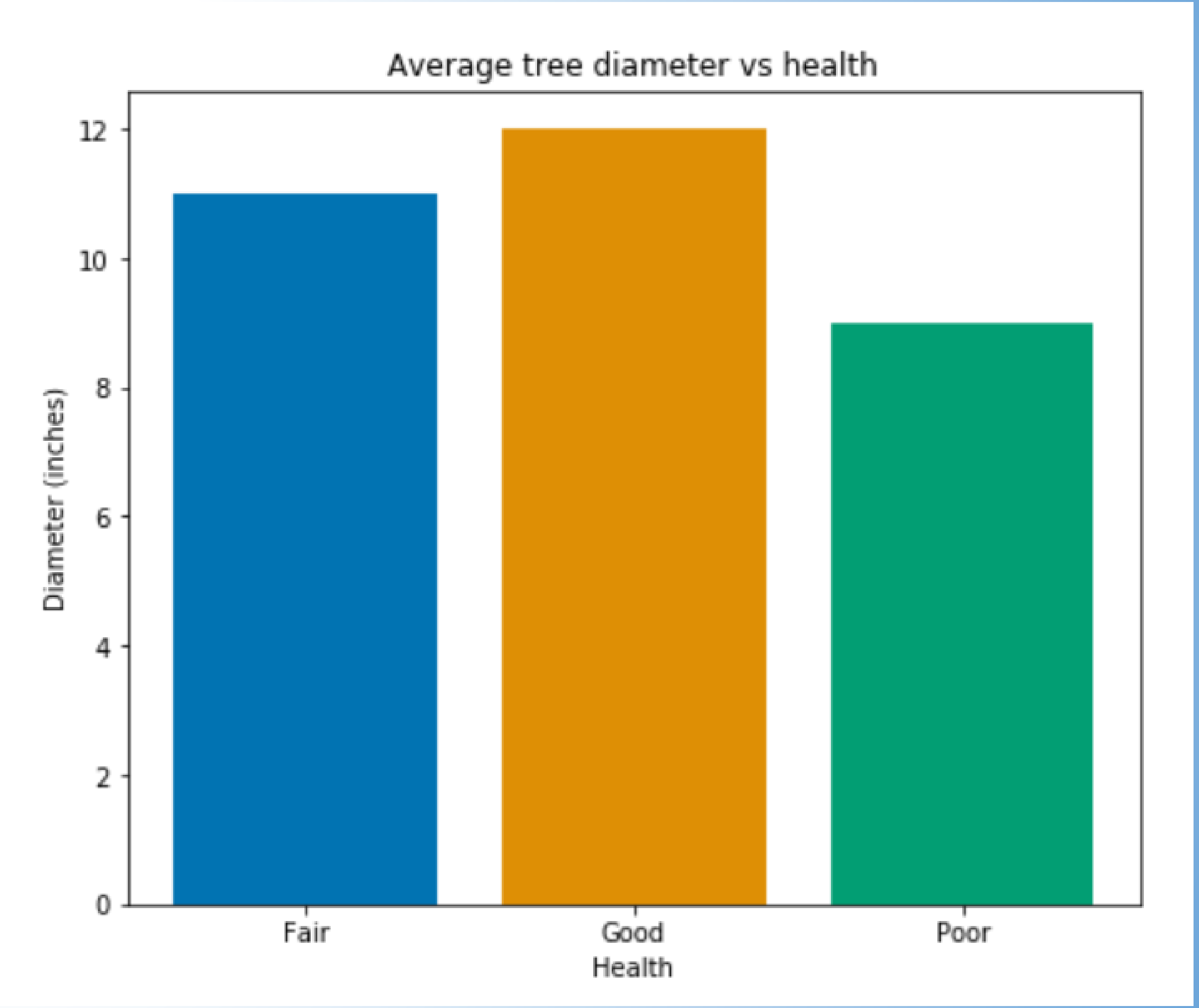
Balancing techniques or weights in the models



EDA – Health

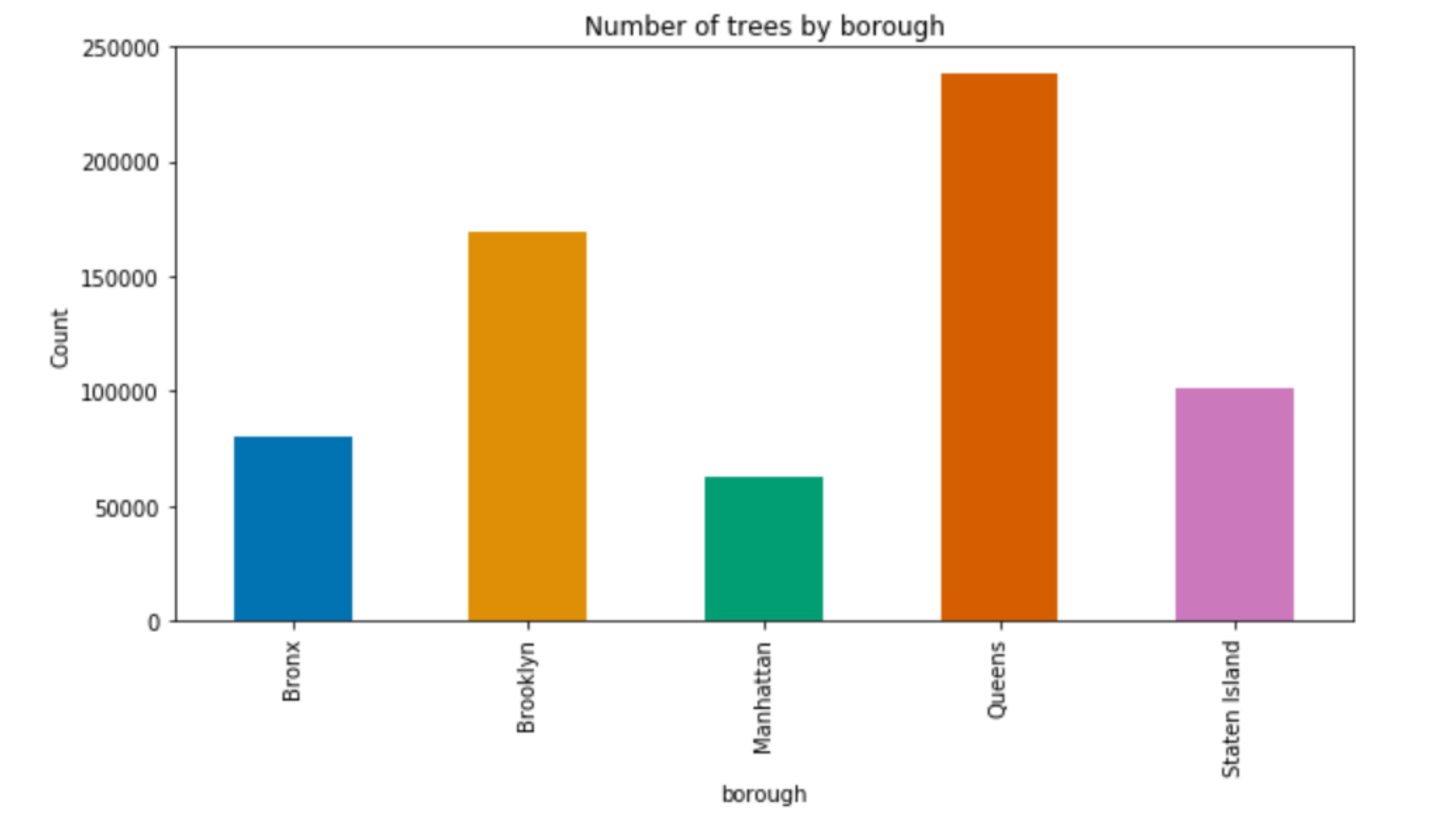
Healthier trees have larger diameter

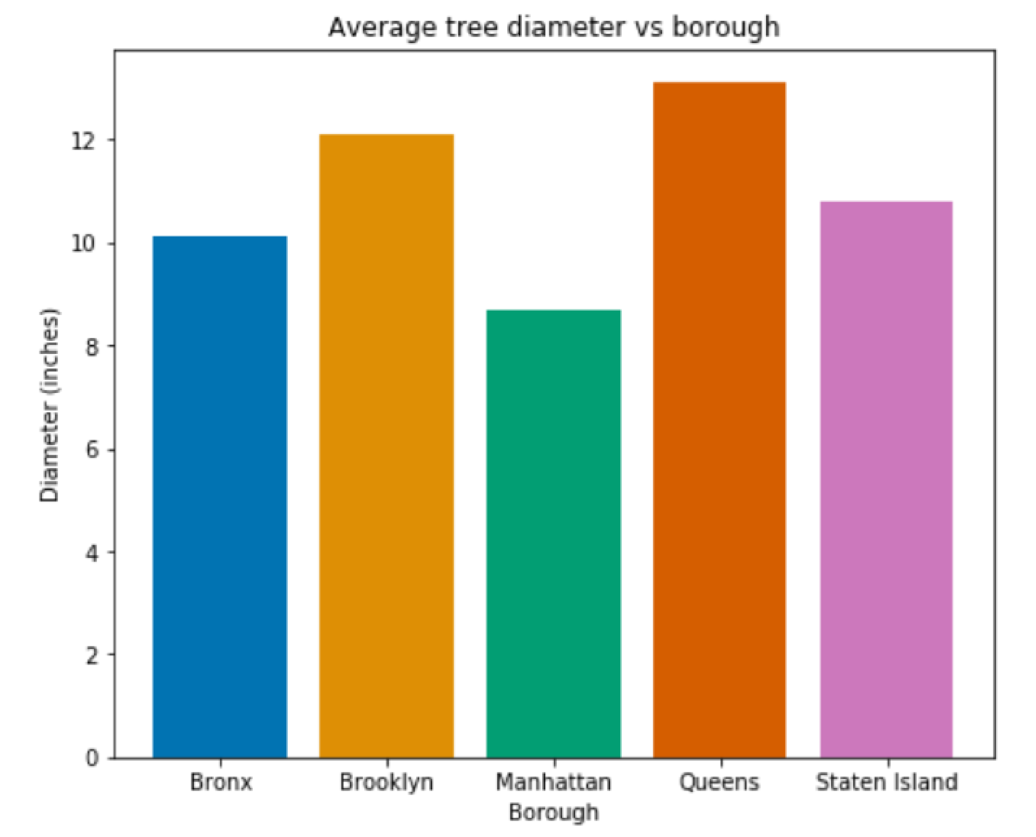
The diameter is a feature that can be used in the model



EDA – Tree Diameter

Where there are more trees, their diameter is bigger

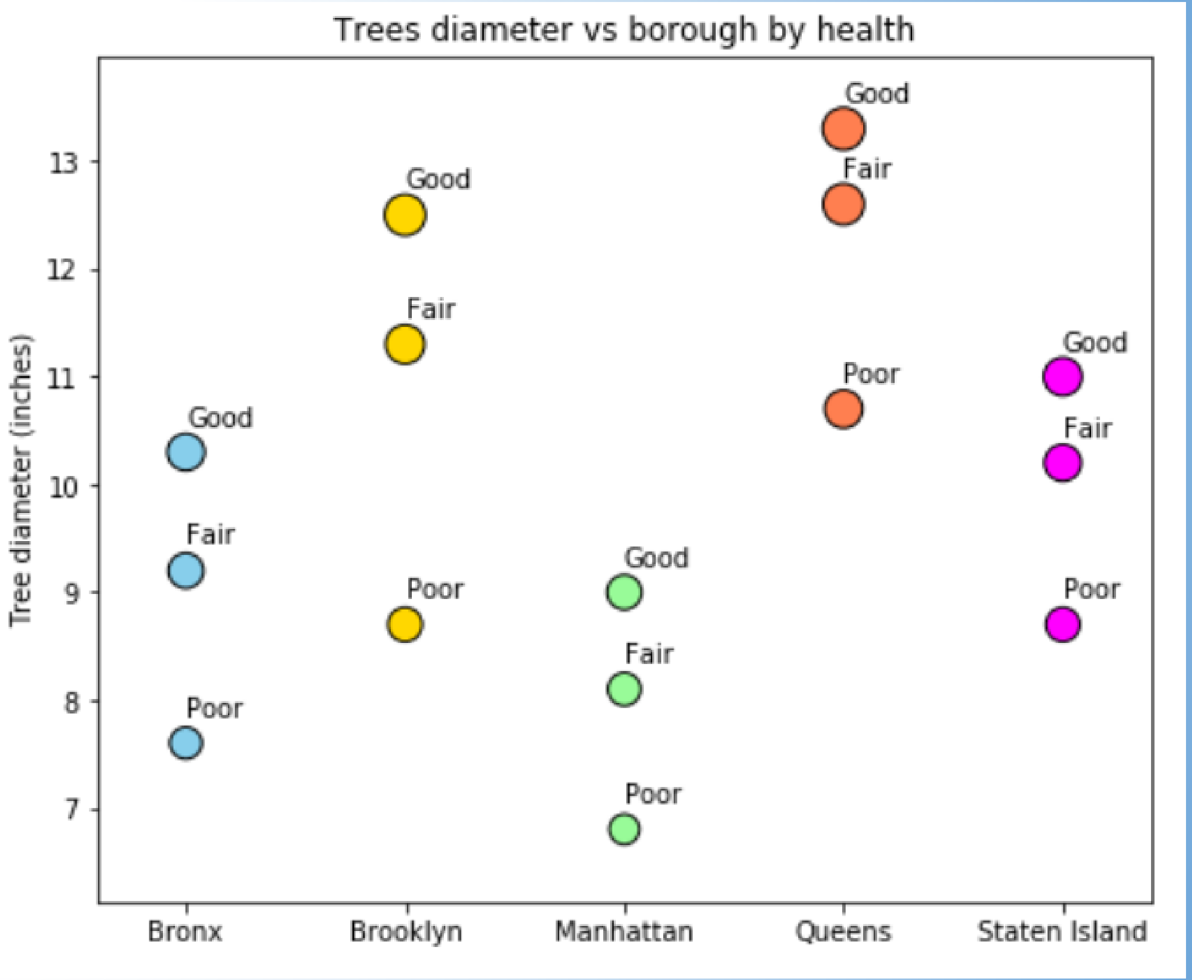




EDA – Tree Diameter

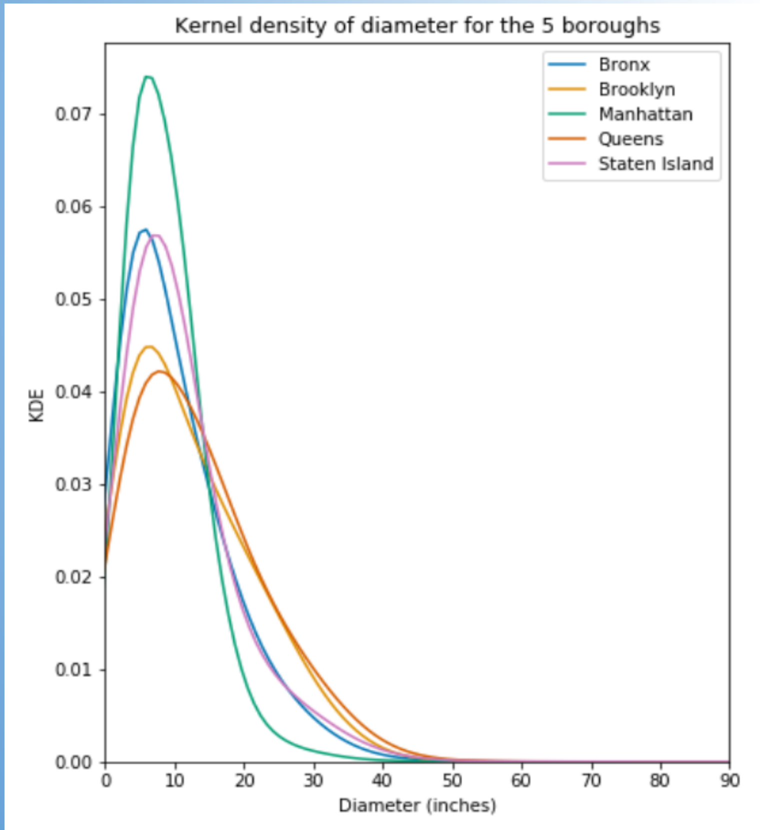
Trees with similar diameter but locates in different areas are classified differently

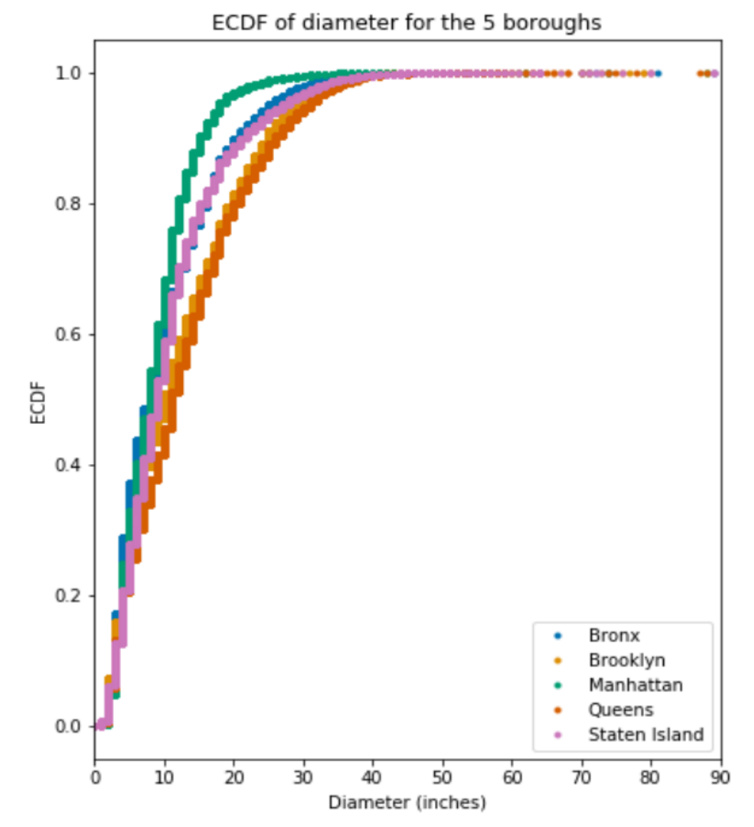
Location is important, the model is fit separately for each borough or the borough information is included if only one model is generated



Statistical Analysis

The trees diameter isn’t normally distributed, and the distribution depends on the location

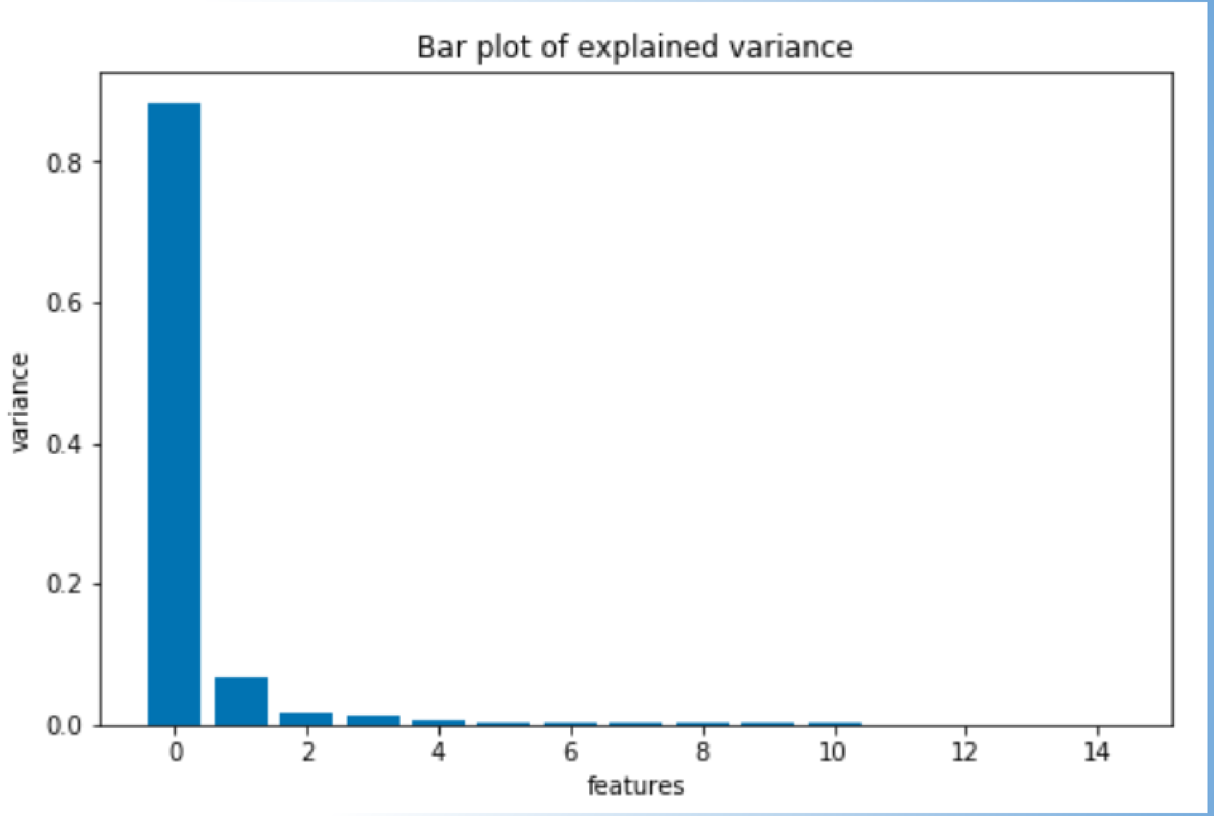




Predictive Model – Features Selection

Various methods used to select features (PCA, SelectKbest, RFE)

The tree diameter is the most important feature, but it isn't enough, so other features are also used



Predictive Model – Description

The variable to predict is the tree health (fair, good, poor) -> multi-class classification problem

Random Forest and K-Nearest Neighbors algorithms are considered to build a model

The models are fit using different predictors (the diameter is always included)

The used metrics are precision, recall and f1 score, ROC and AUC are also evaluated (macro average)

Different classifiers are trained for the different boroughs, or only one classifier is used, but in this case the location information is also included as a feature

Predictive Model – Development

Various sampling techniques are tested to balance the dataset (custom, SMOTEE & Edited Nearest Neighbour from Python “imblearn” package)

The initial fit gives very good results (too good!) on cross validated training set but not on the test set, the model seems to be overfitting.

Different weights to the classes and thresholds have also been tried, but the results on the test set don’t look great

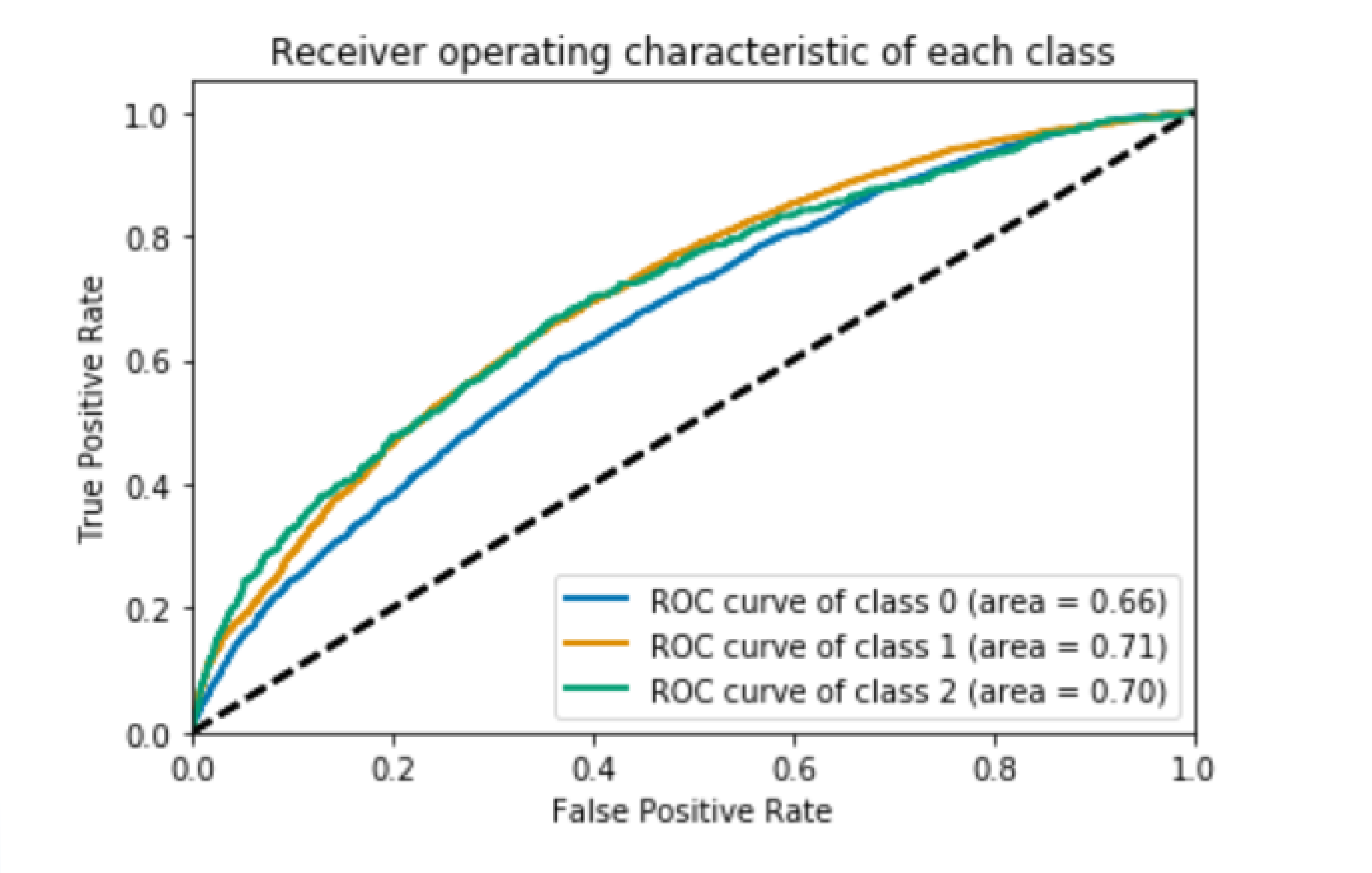
Although features analysis indicates that 2 features at most are relevant, more attempts considering all the encoded variables are made, in order to improve the model (ROC AUC is worse if only 2 features are used)

Predictive Model – ROC Curves (Manhattan)

Random Forest Model

Micro average AUC: 0.78

Macro average AUC: 0.69

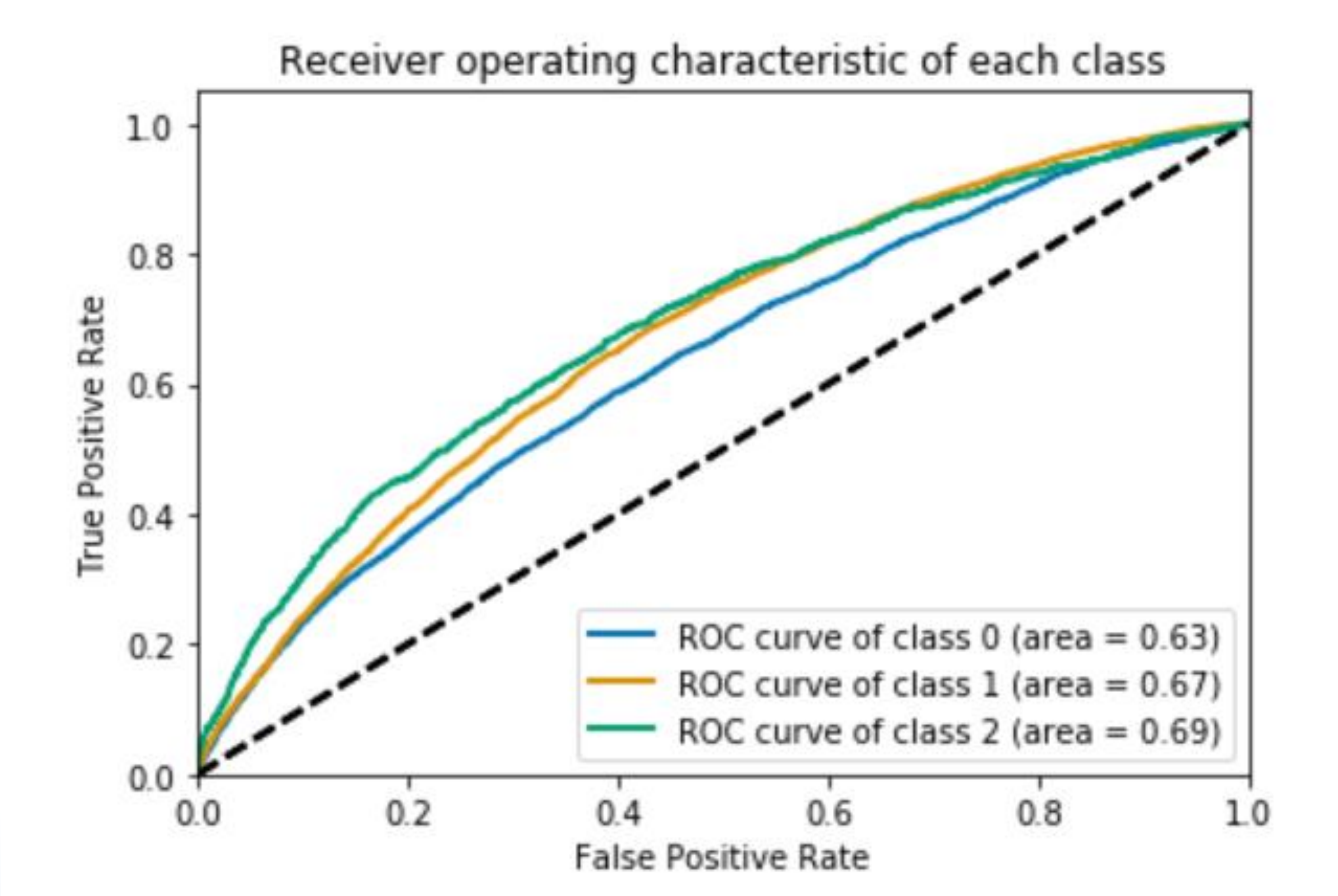


Predictive Model – ROC Curves (Brooklyn)

Random Forest Model

Micro average AUC: 0.74

Macro average AUC: 0.67

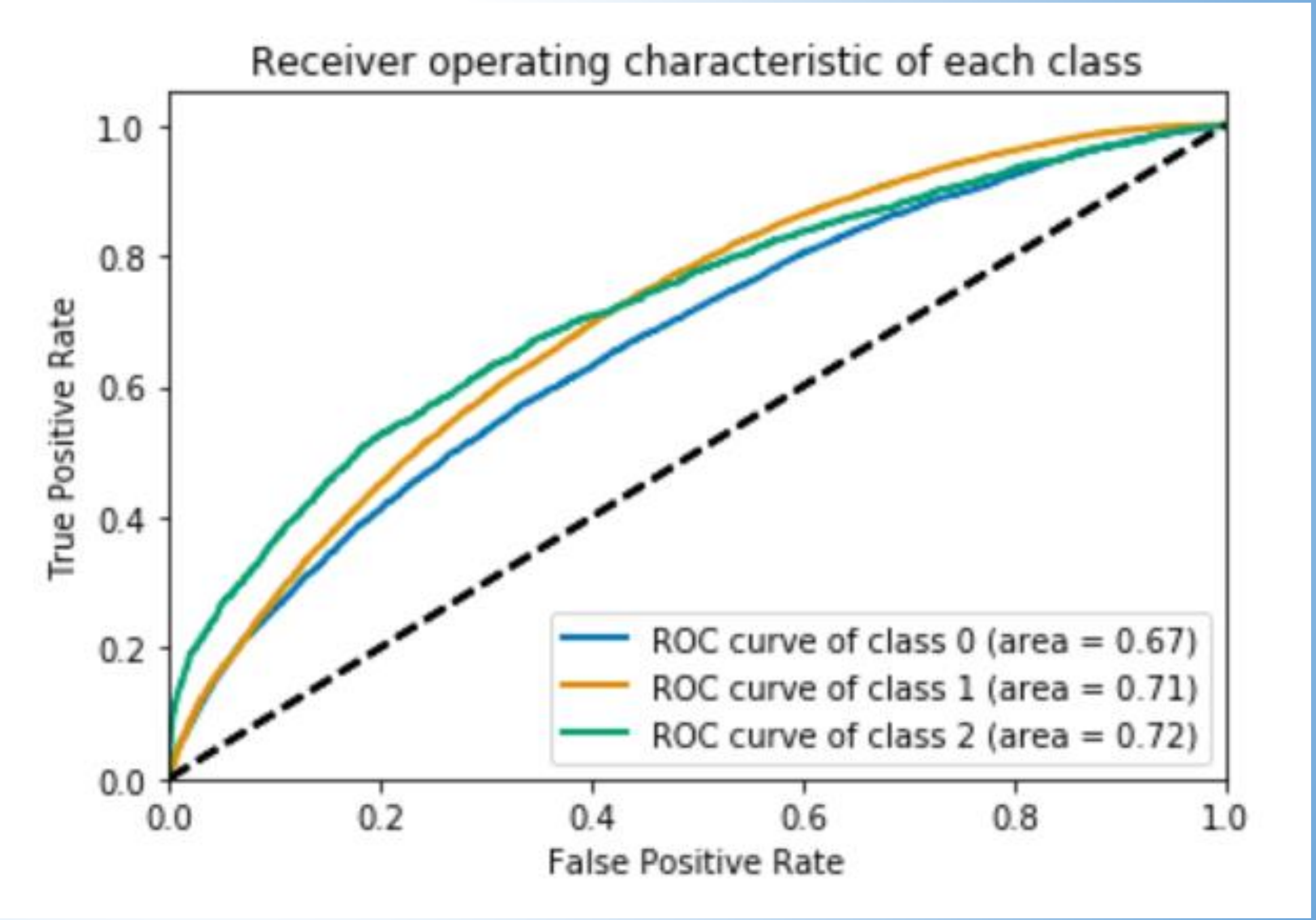


Predictive Model – ROC Curves (Queens)

Random Forest Model

Micro average AUC: 0.80

Macro average AUC: 0.70

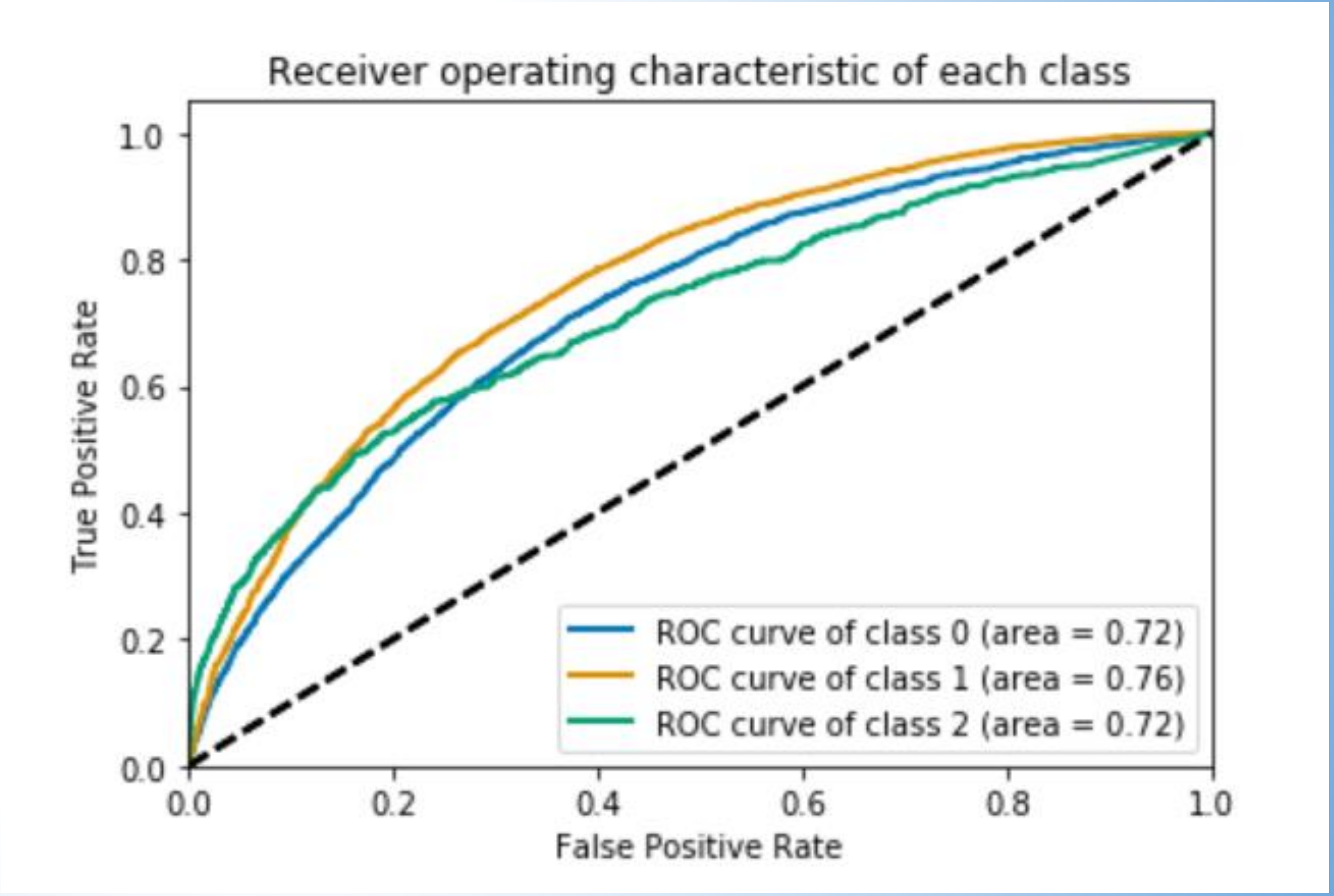


Predictive Model – ROC Curves (Staten Island)

Random Forest Model

Micro average AUC: 0.82

Macro average AUC: 0.73

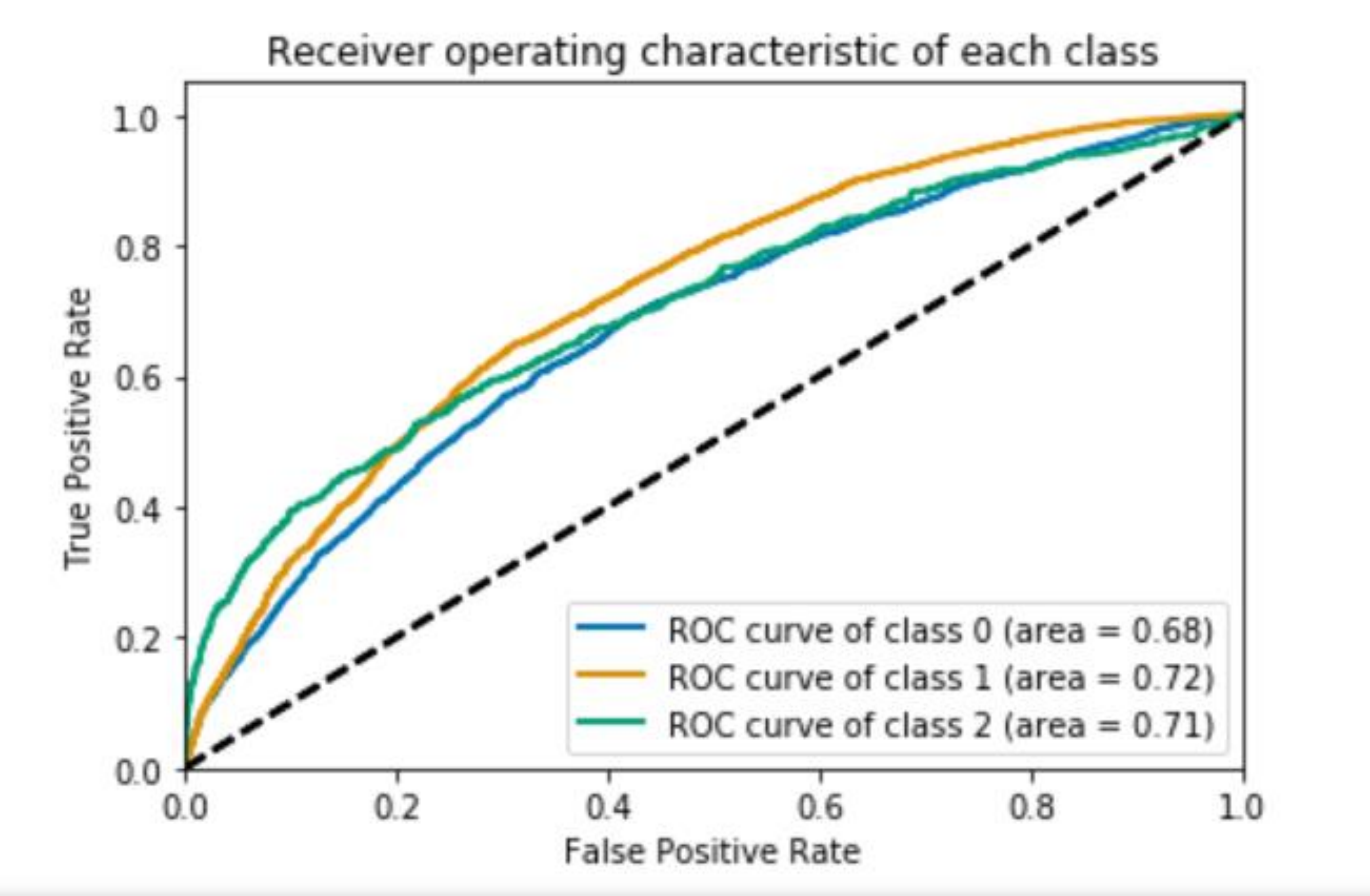


Predictive Model – ROC Curves (Bronx)

Random Forest Model

Micro average AUC: 0.86

Macro average AUC: 0.70

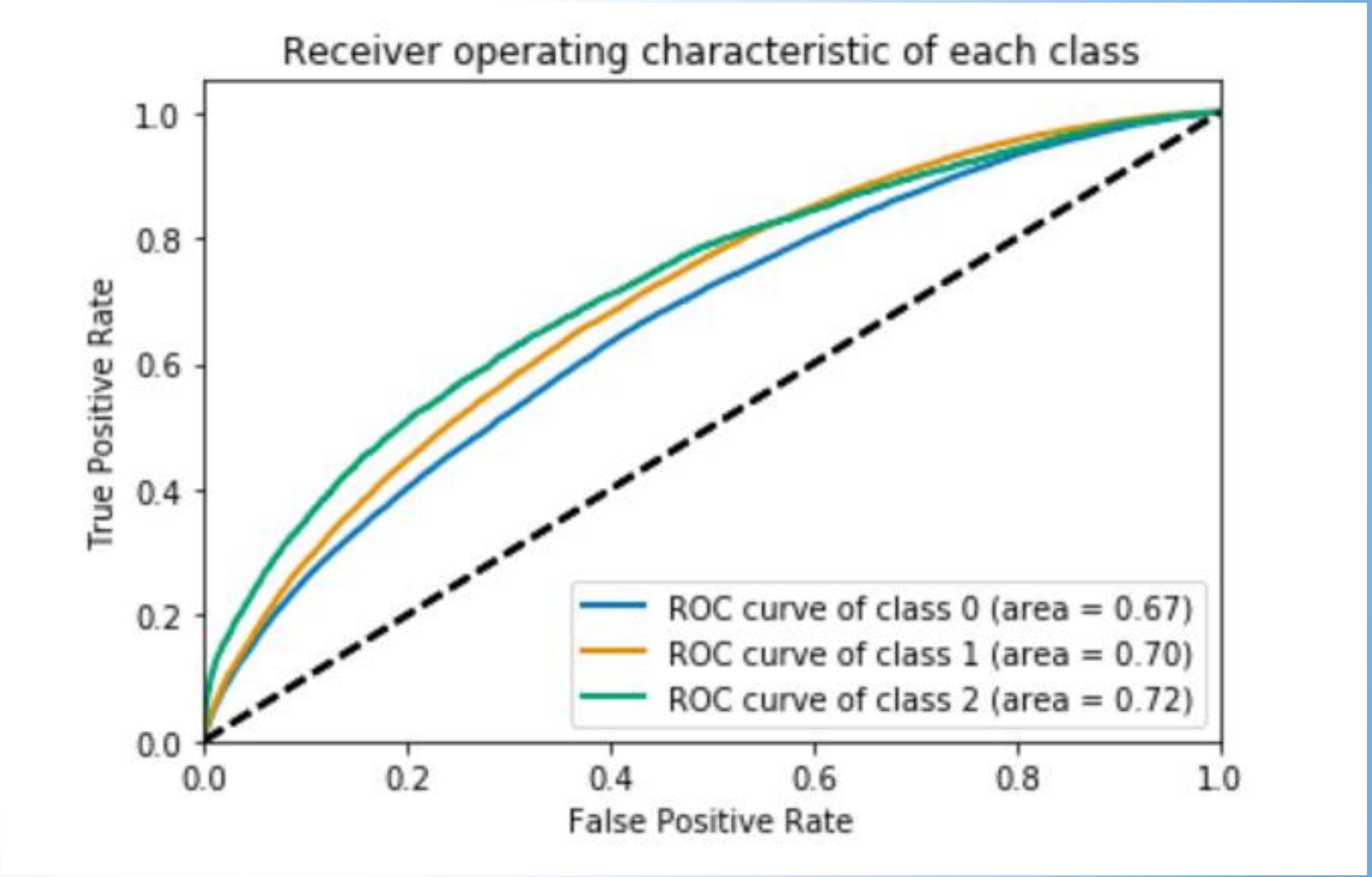


Predictive Model – ROC Curves (all boroughs)

Random Forest Model

Micro average AUC: 0.77

Macro average AUC: 0.70



Model Results – Summary

With the chosen sampler and features, Random Forest performs better then K-Nearest Neighbors

The model can't predict the minority classes with high precision, recall is also low

The best recall case for the "fair" health class is 0.53 in Staten Island and 0.44 in Brooklyn for the "poor" health class

Precision is below 0.3 for the minority classes, above 0.8 for the majority class

Open Questions

Can the features be better selected? Can correlations be better identified?

Can 2 classes be combined?

Can a search be implemented to optimize scores per class (instead of average)?

Should the problem be reformulated as anomaly detection?