

The background image shows a city street scene. In the foreground, there's a dark asphalt road with white lane markings. Several cars are parked along the right side of the road. Behind the cars, there are trees with green foliage. In the background, there are buildings, some with awnings, and a clear blue sky. The overall scene is a typical urban environment.

Predicting Availability of On-Street Parking Spots

Presented by:
Kunal Kotian, Yu Tian

Background Research

stupid puns on parking meters



puns **about** parking

parking **humor**

parking **slogans**

funny parking **quotes**

quotes about finding parking

parking **lot humor**

parking **one liners**

14



The Parking Meteors



0.54611

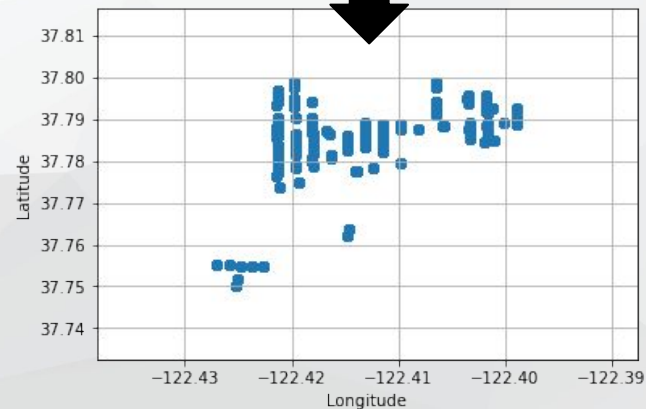
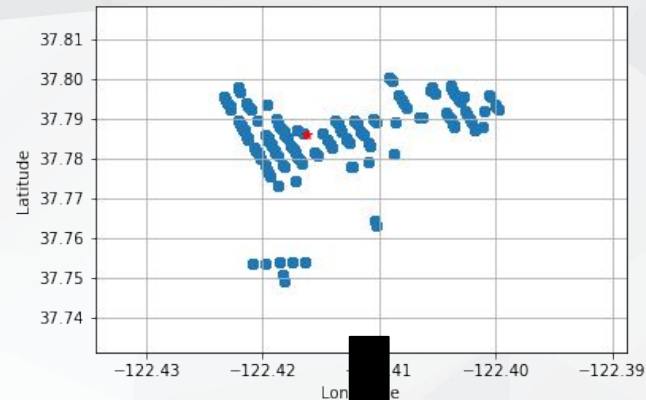
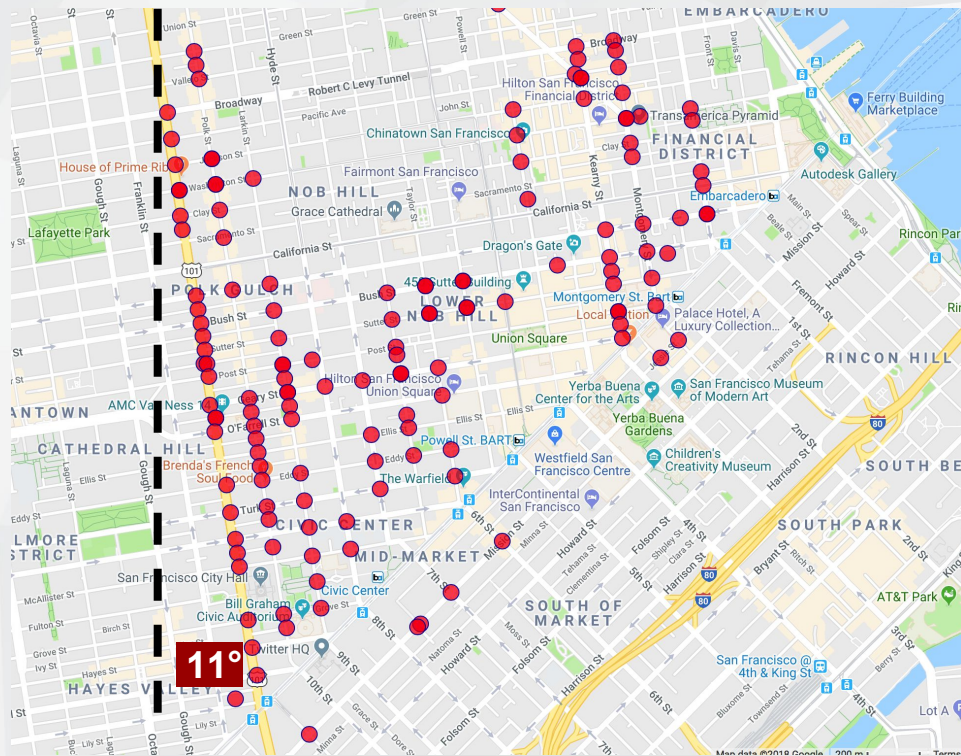
16

5h

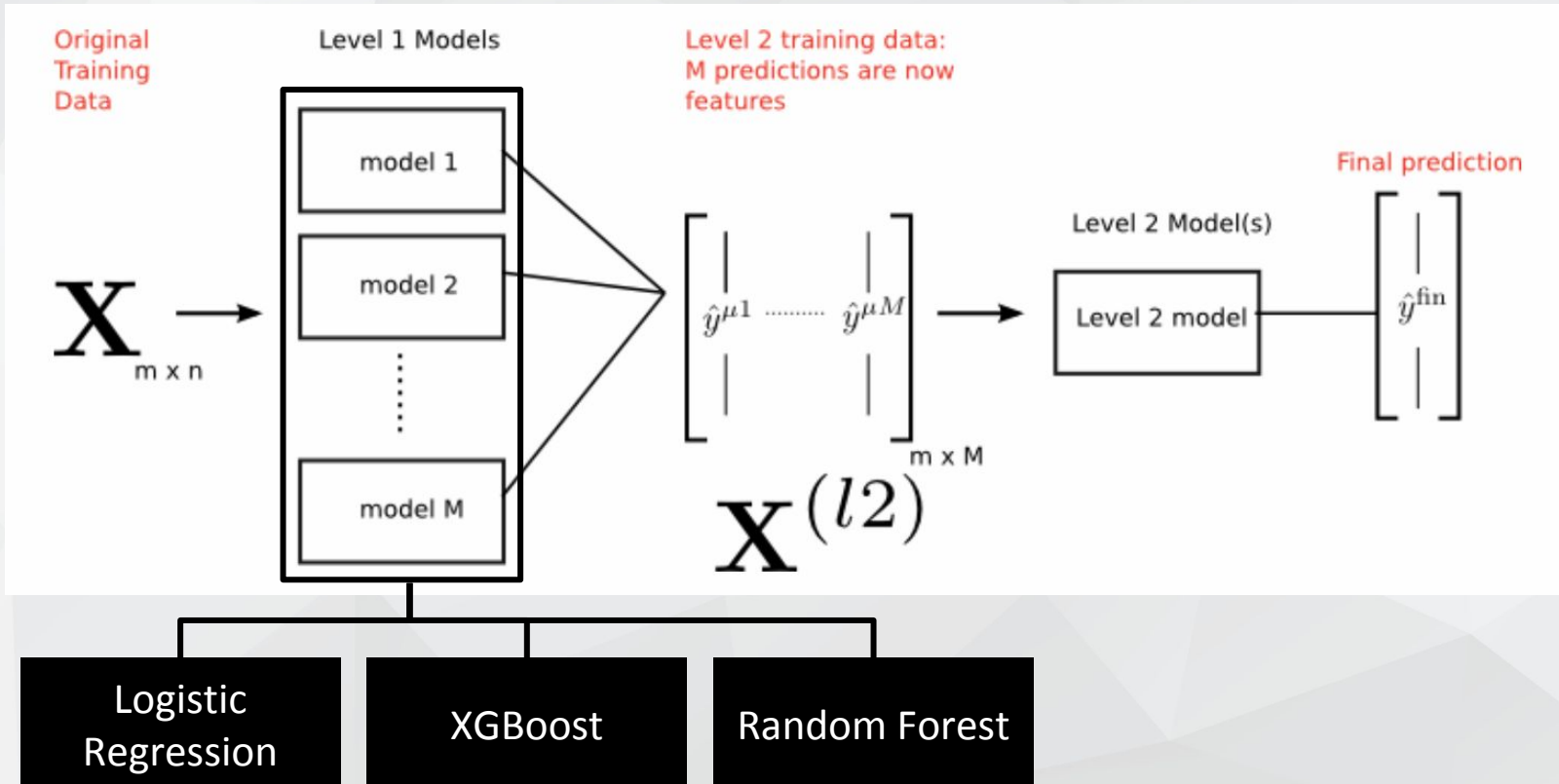
Feature Engineering (1/2)

- Most important features: simple target encoding
- Datetime
 - 'day_part': morning/afternoon/evening/night
 - Target encoded features based on day_part
 - 'minutes_since_midnight': high importance for RF + XGBoost
- Parkingrecords.csv
 - Mapped meters to street segments in training data
 - Several mean encoded features using transactions per street segment

Feature Engineering (2/2)



Machine Learning



Mlxtend: Concise Code for Stacking Grid Search

```
clf1 = LogisticRegression()
clf2 = RandomForestClassifier(n_estimators=500, max_features='sqrt')
clf3 = XGBClassifier()
lr = LogisticRegression()
sclf = StackingClassifier(classifiers=[clf1, clf2, clf3], meta_classifier=lr)


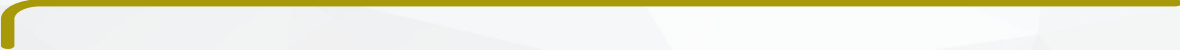

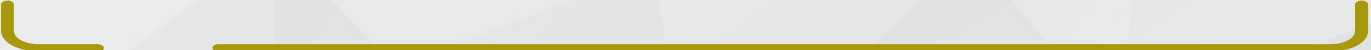
params = {'logisticregression__C': [1, 10],
          'randomforestclassifier__min_samples_leaf': [10, 25],
          'xgbclassifier__n_estimators': [20, 40, 60],
          'xgbclassifier__learning_rate': [0.001, 0.01, 0.1],
          'xgbclassifier__subsample': [0.5, 0.75],
          'xgbclassifier__max_depth': [3, 5, 10],
          'xgbclassifier__colsample_bytree': [0.3, 0.5],
          'meta-logisticregression__C': [10, 100]}

grid = GridSearchCV(estimator=sclf, param_grid=params, cv=5, scoring=fhalf_scorer, refit=True, n_jobs=-1, verbose=1)
grid.fit(X, y)
```



Results & Conclusions

F0.5 Score	Precision	Recall
0.58	0.58	0.58

- 
- 
- 
1. Starting with simple models and available data
 2. Importance of background research
 3. Benefits of a well-designed workflow
- 
- 