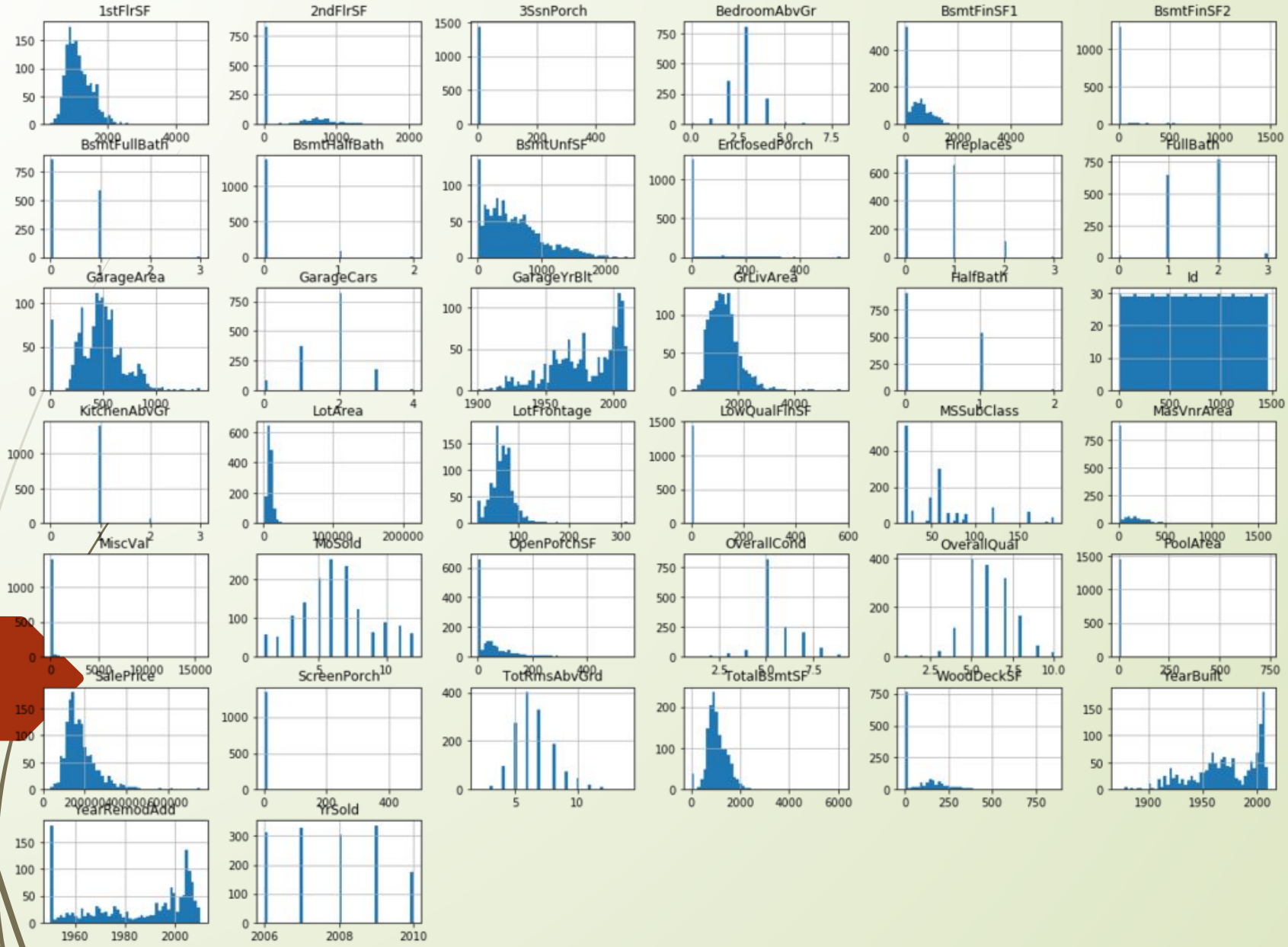
Kaggle Competition

House Prices: Advanced Regression Techniques

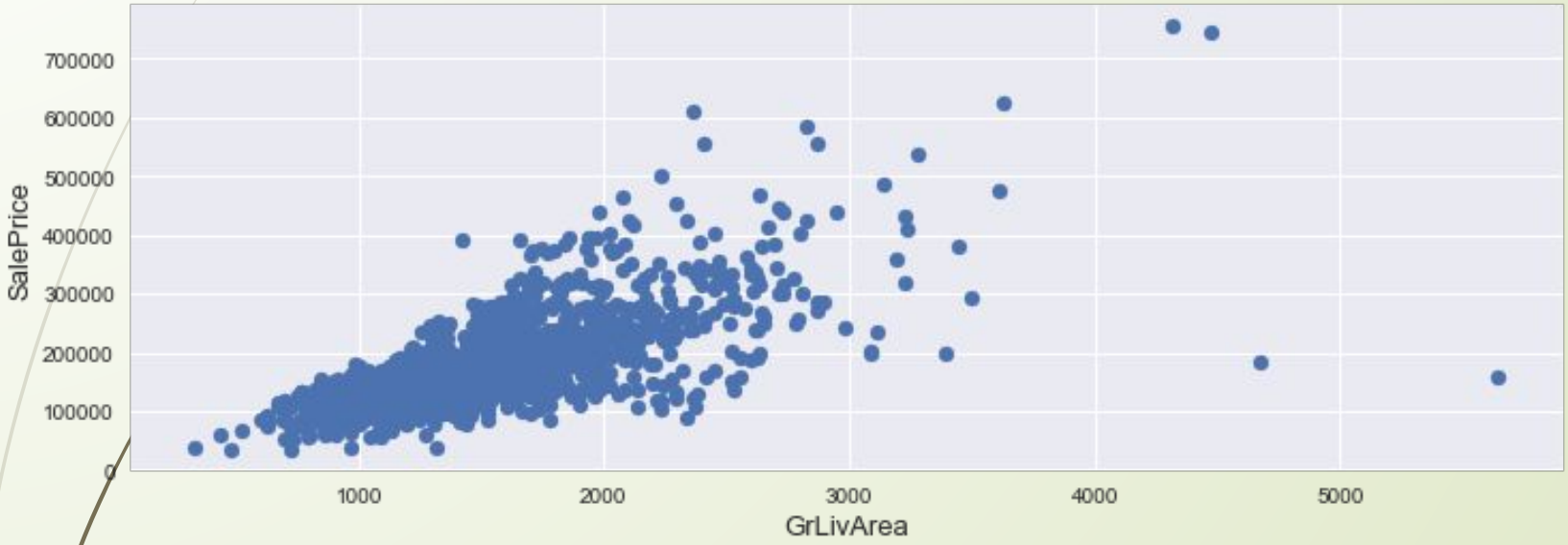
David | Neuton | Shubh | Vineet


1stFlrSF
2ndFlrSF
3SsnPorch
BedroomAbvGr
BsmtFinSF1
BsmtFinSF2
BsmtFullBath
BsmtHalfBath
BsmtUnfSF
EnclosedPorch
Fireplaces
FullBath
GarageArea
GarageCars
GarageYrBlt
GrLivArea
HalfBath
Id
KitchenAbvGr
LotArea
LotFrontage
LowQualFinSF
MSSubClass
MasVnrArea
MiscVal
MoSold
OpenPorchSF
OverallCond
OverallQual
PoolArea
SalePrice
ScreenPorch
TotRmsAbvGrd
TotalBsmtSF
WoodDeckSF
YearBuilt
YearRemodAdd
YrSold

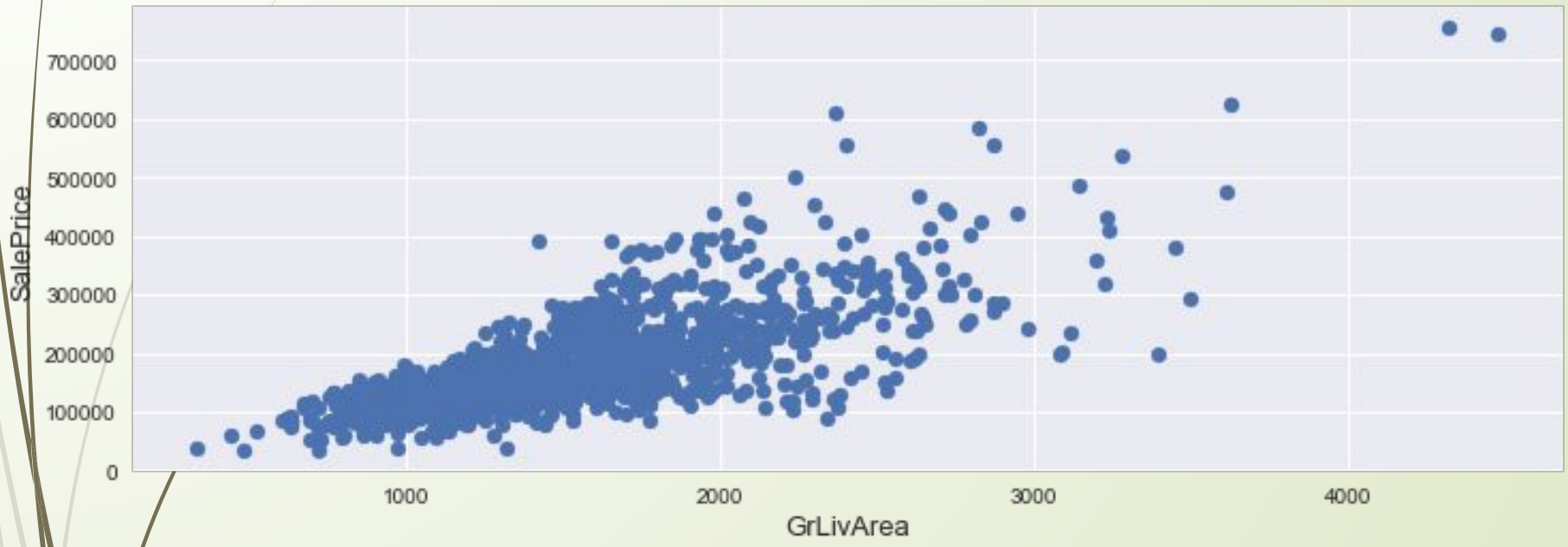
Preprocessing

- Irreducible Error:
  - Missing Values
  - Outliers
- Variance:
  - Normalizing
  - Removing Columns
  - Adding Features
- Bias:
  - Transformed features

Outliers


700000
600000
500000
400000
300000
200000
100000
0
SalePrice
1000
2000
3000
4000
5000
GrLivArea

Outliers


SalePrice
700000
600000
500000
400000
300000
200000
100000
0
1000
2000
3000
4000
GrLivArea

Missing Values

```
all_data["Alley"] = all_data["Alley"].fillna("None")
all_data["Fence"] = all_data["Fence"].fillna("None")
all_data["FireplaceQu"] = all_data["FireplaceQu"].fillna("None")
for col in ('GarageType', 'GarageFinish', 'GarageQual', 'GarageCond'):
    all_data[col] = all_data[col].fillna('None')
all_data["MasVnrType"] = all_data["MasVnrType"].fillna("None")
for col in ('BsmtQual', 'BsmtCond', 'BsmtExposure', 'BsmtFinType1', 'BsmtFinType2'):
    all_data[col] = all_data[col].fillna('None')
all_data["Functional"] = all_data["Functional"].fillna("Typ")
all_data['MSSubClass'] = all_data['MSSubClass'].fillna("None")

all_data["LotFrontage"] = all_data["LotFrontage"].fillna(0)
for col in ('GarageYrBlt','GarageArea', 'GarageCars'):
    all_data[col] = all_data[col].fillna(0)
for col in ('BsmtFinSF1', 'BsmtFinSF2', 'BsmtUnfSF','TotalBsmtSF', 'BsmtFullBath', 'BsmtHalfBath'):
    all_data[col] = all_data[col].fillna(0)
all_data["MasVnrArea"] = all_data["MasVnrArea"].fillna(0)

all_data['MSZoning'] = all_data['MSZoning'].fillna(all_data['MSZoning'].mode()[0])
all_data['Electrical'] = all_data['Electrical'].fillna(all_data['Electrical'].mode()[0])
all_data['KitchenQual'] = all_data['KitchenQual'].fillna(all_data['KitchenQual'].mode()[0])
all_data['Exterior1st'] = all_data['Exterior1st'].fillna(all_data['Exterior1st'].mode()[0])
all_data['Exterior2nd'] = all_data['Exterior2nd'].fillna(all_data['Exterior2nd'].mode()[0])
all_data['SaleType'] = all_data['SaleType'].fillna(all_data['SaleType'].mode()[0])
```

Examples

Numeric Features:

- Removed 2 outliers on *GrLivArea* (Above grade (ground) living area square feet) feature;
- Transformed *GrLivArea* using Square Root;
- Transformed *TotalBsmtSF* using Cubic Root;
- Transformed the Bathrooms features where half bathroom was inputted with .5 and full bathroom with 1;
- Transformed *LotArea* with Log;
- Transformed *LotFrontage* using Square Root;
- Created *ShedSF* based on the *MiscValue* feature;
- Dropped some features with multicollinearity.

Examples

Categorical Features:

- Joined *BasementType* 1 and 2;
- Joined *ExternalMaterial*;
- Created a "Has Fence" feature;
- Joined *Fireplaces* and *Fireplace Quality*;
- Transformed *YearRemodAdd* into range of years;
- Dropped some features with more than 90% of NA or very high frequency of the same category.

Categorical Changes

```
#Selecting columns to apply LabelEncoder
from sklearn.preprocessing import LabelEncoder
cols = ('GarageCond','FireplaceQu', 'BsmtQual', 'BsmtCond',
        'ExterQual', 'ExterCond','HeatingQC',
        'KitchenQual', 'Functional', 'BsmtFinType1', 'BsmtFinType2',
        'BsmtExposure', 'GarageFinish', 'LandSlope',
        'LotShape', 'PavedDrive',
        'CentralAir', 'OverallCond',
        'YrSold')
#Process columns, apply LabelEncoder to categorical features
for c in cols:
    lbl = LabelEncoder()
    lbl.fit(list(all_data[c].values))
    all_data[c] = lbl.transform(list(all_data[c].values))
```

```
#Getting dummies for categorical features
all_data = pd.get_dummies(all_data)
```

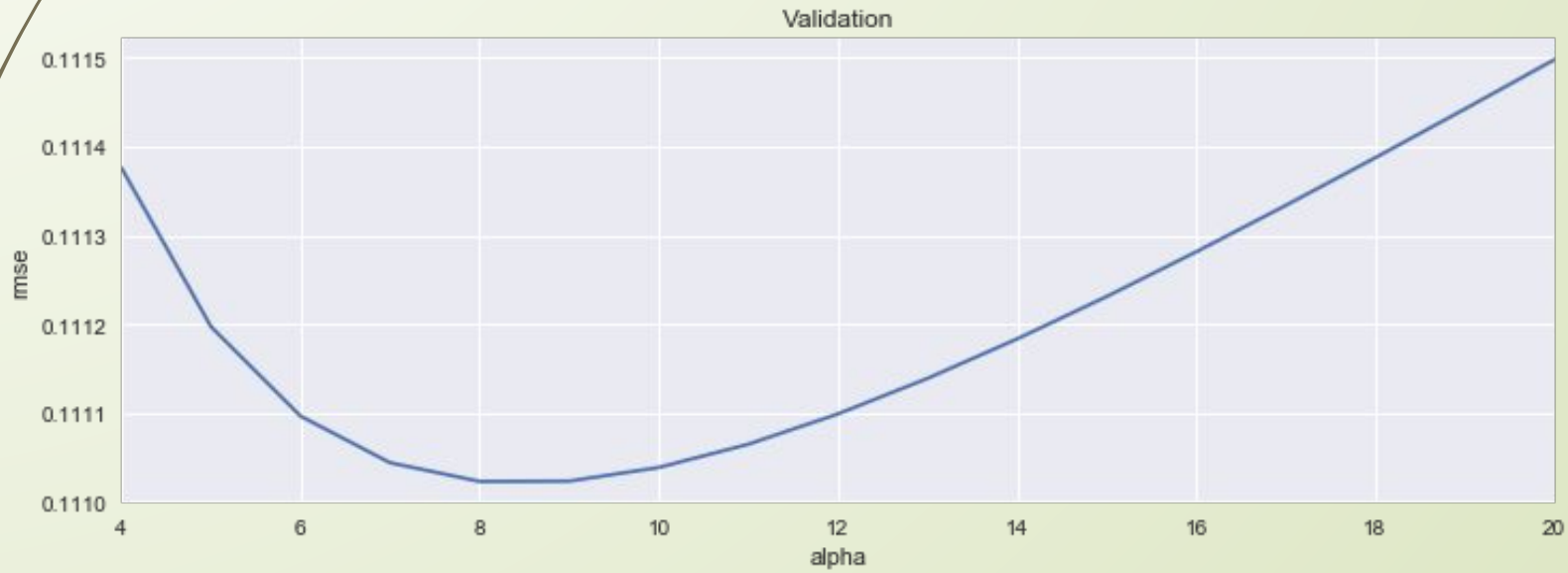
MODELING

Linear Modeling

- Lasso
  - Alpha: 0.0005
  - Cross Val Score: 0.1090
  - Kaggle: 0.117890
- Ridge
  - Alpha: 8
  - Cross Val Score: 0.1109
  - Kaggle: 0.12569
- Elastic Net
  - Alpha: .0006
  - l1 ratio: .8
  - cross val: 0.1091
  - Kaggle: 0.11784

An Example of Parameters Selections

We plotted a range of alpha parameters against the RMSE to find the lowest RMSE for Lasso.



Other Modeling

- Support Vector Regression
  - Kernal: ‘RBF’
  - C: 10
  - gamma: .001
  - epsilon: 0.0001
  - Cross Val Score: 0.129
  - BoxCox Transformation
  - Kaggle: 0.128

- Random Forest
  - N_estimators = 100
  - min sample split = 2
  - min sample leaf = 1
  - cross val score = 0.26316

Gradient Boosting

More generic Feature engineering

- n_estimators = 3000'
- learning_rate = 0.0465
- max_depth = 2
- max _features = 'sqrt'
- min_samples_leaf = 3
- min_samples_split = 18
- loss = 'huber'
- cross val score = 0.1093
- Kaggle Score = 0.12394

Scaling

Robust Scaling vs MinMax Scaling

$$\frac{x_i - Q_1(x)}{Q_3(x) - Q_1(x)}$$

$$\frac{x_i - min(x)}{max(x) - min(x)}$$

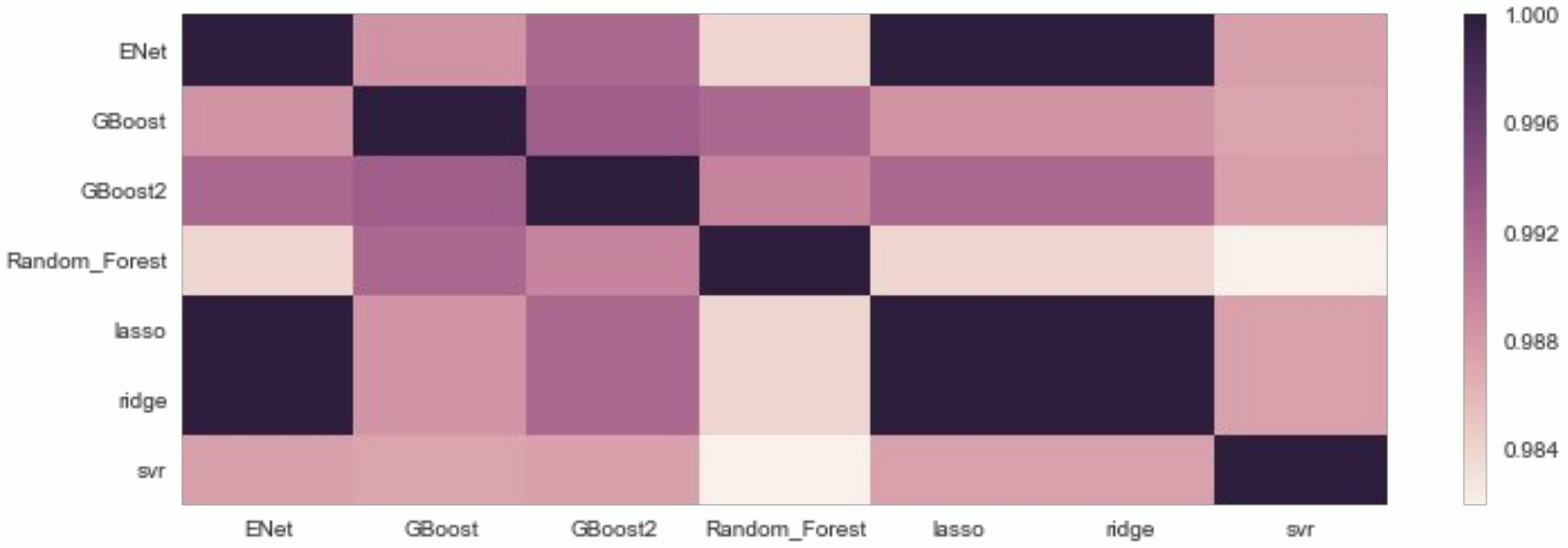
Hyperparameters

- Grid Search
- Bayesian Optimization
- Graphing
- Trial/Error

What	Ridge	Alpha	What	Lasso	Alpha
No change, only log and scale	0.143898	15	No change, with scale no log	30842.06271	140
Drop 1stFlrSF	0.14391	15	Drop 1stFlrSF	30842.24917	141
Drop 2ndFlrSF	0.143876	15	Drop 2ndFlrSF	30832.50556	142
Drop LowQualFinSF	0.143578	15	Drop LowQualFinSF	30914.9163	154
Sqrt of GrLivArea	0.139058	13	Sqrt of GrLivArea	30707.68815	148
Transforming BedroomAbvGr	0.139118	13	Transforming BedroomAbvGr	30698.94186	148
Dropping BsmtFinSF1	0.138869	14	Dropping BsmtFinSF1	30691.91287	148
Dropping BsmtFinSF2	0.13888	14	Dropping BsmtFinSF2	30672.54849	148
Dropping BsmtUnfSF	0.137799	14	Dropping BsmtUnfSF	30405.52301	147
Transforming TotalBsmtSF	0.135818	14	Creating TotalBsmtSF	29855.16032	148
Dropping BsmtFinSF2 after transforming TotalBsmtSF	0.135759	14	Dropping BsmtFinSF2 after creating TotalBsmtSF	29845.56901	148
Transforming BsmtFullBath	0.135837	14	Transforming BsmtFullBath	29826.60025	147
Dropping BsmtHalfBath	0.135715	14	Dropping BsmtHalfBath	29837.92093	148
Creating TotalPorchSF	0.135751	14	Creating TotalPorchSF	29840.19761	132
Dropping EnclosedPorch	0.13582	15	Drop EnclosedPorch	29830.38721	148
Dropping OpenPorchSF	0.13542	14	Drop OpenPorchSF	29794.306	148
Dropping ScreenPorch	0.136705	15	Dropping ScreenPorch	29883.75077	131
Dropping 3SsnPorch	0.135351	14	Dropping 3SsnPorch	29782.94176	135
Changing Fireplaces	0.135351	14	Changing Fireplaces	29782.95679	135
Transforming bathrooms	0.135059	14	Transforming bathrooms	29706.91771	136
Dropping GarageArea	0.134446	14	Dropping GarageArea	29541.78107	134
Dropping KitchenAbvGr	0.134951	14	Dropping KitchenAbvGr	29716.47395	135
Transforming LotArea	0.13355	**15**	Transforming LotArea	29513.1328	148
Transforming LotFrontage	0.133404	15	Transforming LotFrontage	29492.28497	150
Dropping MasVnrArea	0.133182	14	Dropping MasVnrArea	29384.08751	155
Creating Shed	0.132871	14	Creating Shed	29379.31959	155
Dropping PoolArea	0.132933	14	Dropping PoolArea	29365.91545	157
Dropping TotRmsAbvGrd	0.132636	15	Dropping TotRmsAbvGrd	29392.77119	154
Transforming WoodDeckSF	0.132719	15	Transforming WoodDeckSF	29412.51672	158
Dropping WoodDeckSF	0.133441	15	Dropping WoodDeckSF	NA	NA
Removing outliers	0.132491	15	Removing outliers	28841.1395	149

Model	CV Score	Std	Feature Engineering Changes
Gradient Boosting	0.1199	0.0129	keep outliers
Gradient Boosting	0.1126	0.0068	remove 2 outliers
Gradient Boosting	0.1118	0.0078	drop poolQC
Gradient Boosting	0.112	0.0066	drop poolQC and fence
Gradient Boosting	0.1112	0.0066	drop poolQC and MiscFeatures
Gradient Boosting	0.1122	0.007	drop poolQC and MiscFeatures and Alley
Gradient Boosting	0.1108	0.006	lot frontage na = 0
Gradient Boosting	0.1112	0.0066	lot frontage na = median of neighborhoods
Gradient Boosting	0.1111	0.0067	lot frontage na = mean
Gradient Boosting	0.1123	0.007	GarageYrBuilt converted to hasGarage Yes or No
Gradient Boosting	0.1124	0.0071	Drop BsmtFinSF1
Gradient Boosting	0.1126	0.0069	Did not add TotalBsmtSF, 1stFlrSF, 2ndFlrSF
Gradient Boosting	0.1112	0.0066	remove grlivarea>4000 outliers
Gradient Boosting	0.1113	0.0059	remove index 1128 as outlier

Correlation of the Models



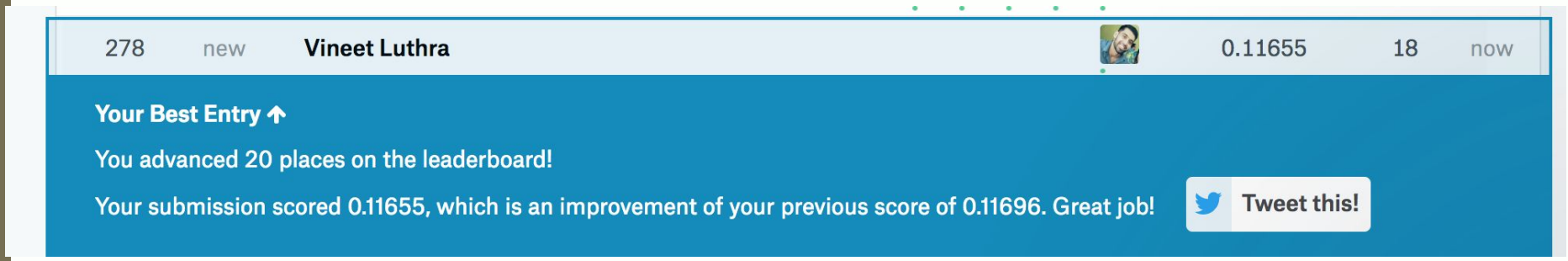
Combining Models

- Average
- Weighted Average
- Stacking

Combining Models

kaggle

(GB + Lasso + Ridge + EN + SVR) / 5



Kaggle Score: 0.11655

THANK YOU