Home Price Prediction in Ames, Iowa – Team 84

Team Members:

1. Reena Sahai: (RSahai6)
2. Jonathan Gerszberg (JGerszberg3)
3. Vignesh Thavamani Thenmozhi (VThenmozhi3)
4. Siddharth Gudiduri (SGudiduri3)

Georgia Tech

Team Members

1. Reena Sahai: (RSahai6)

Reena Sahai is a data quality analyst at a renowned US national bank with 14 years of experience working as a database developer and reporting analyst in various domains like education, healthcare, and finance. She got a Master's degree in Computer science.

2. Jonathan Gerszberg (JGerszberg3)

Jonathan is a software developer at Siemens in Ann Arbor, MI. He got a BS at Rutgers University in Math and chemical engineering, and an MS at the University of Michigan in chemical engineering.

3. Vignesh Thavamani Thenmozhi (VThenmozhi3)

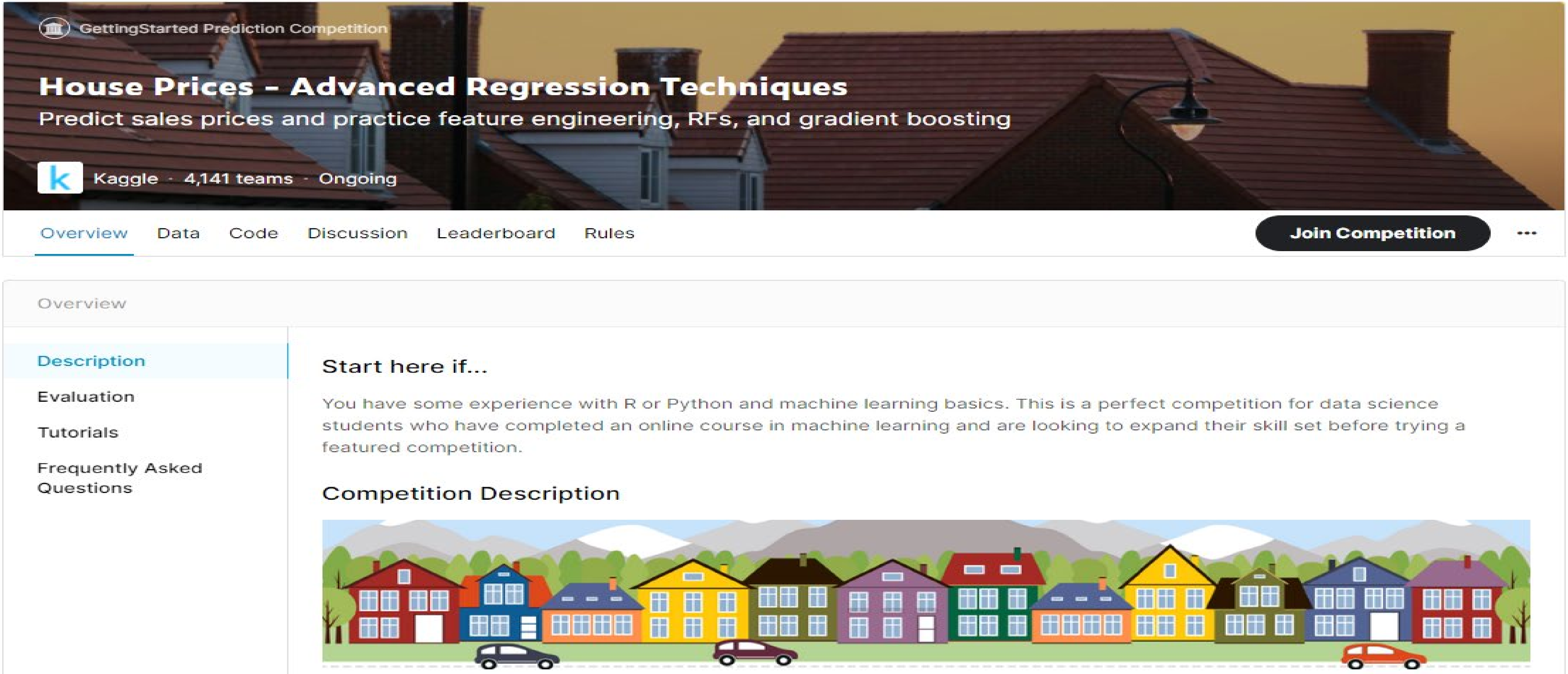
Vignesh is based out of Atlanta and has extensive experience in the Retail and Supply Chain Industry. He has a bachelor's in Electrical and Electronics Engineering and a diploma in Business Management.

4. Siddharth Gudiduri (SGudiduri3)

Siddharth is an experienced engineer with a strong background in computer science and with 15 years of experience in Software Development Lifecycle (SDLC). He has solid hands-on experience from redesigning existing software to solving complex technical problems with an emphasis on object-oriented design and programming.

Georgia Tech

Ongoing Kaggle Competition



Problem Statement

Homes, that are geographically close, can have significantly varied prices. This research aims to gain key insight into what physical and environmental aspects of a home make it more or less expensive. With this understanding, it is the hope that a model can be created to **predict the price** of the home. This investigation will primarily focus on the Ames housing dataset, which was compiled by Dean De Cock, and the environment data from, the website, *Neighborhood Scout*.

The purpose of this project is to minimize the Mean Square Error (MSE) of the predicted sales price of the home vs the actual sales price of the home and to maximize $\boldsymbol{R2}$ (explainable variation). Formulating the above problem, some variables, and objective functions can be defined as follows:

Variables

- *let* $a_0, a_1, a_2, \dots a_m \in A$ *be independent variables*
- *let* $x_{ij} \in X$*, be data points, i.e.* j^{th} *factor of data point* i

Objective function

- *Minimize* $\sum_{i=1}^{n}\left(y_i - \left(a_0 + \sum_{j=1}^{m} a_j x_{ij}\right)\right)^2$

Constraint

- *Lasso regression constraint* $\sum_{j=1}^{m}|a_j| \leq T$
- *Ridge regression constraint* $\sum_{j=1}^{m}(a_j)^2 \leq T$

Georgia Tech

Hypothesis

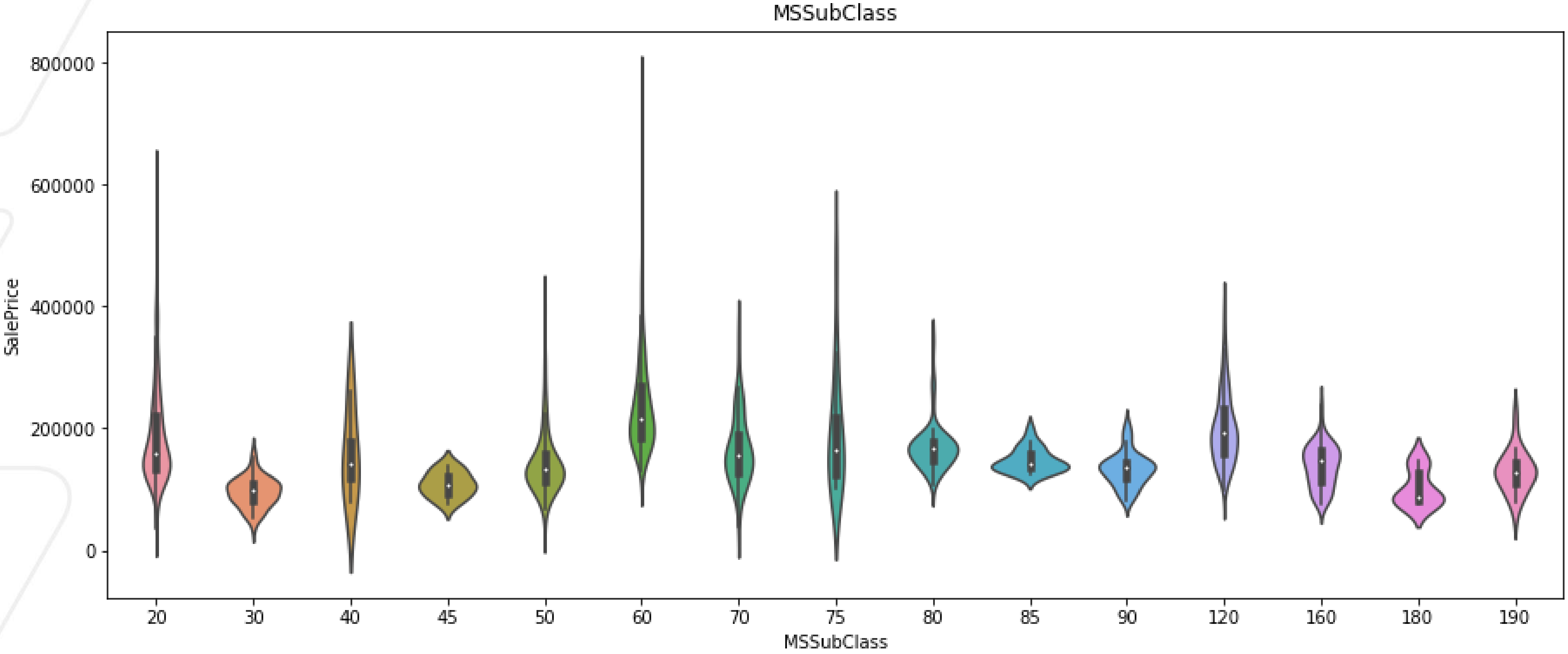
Does there exist a relationship between the dependent variable(Sale Price) and independent variable in dataset ? How does the number of bedroom, lot size and other attribute relate to Sale Price?

Data Analysis

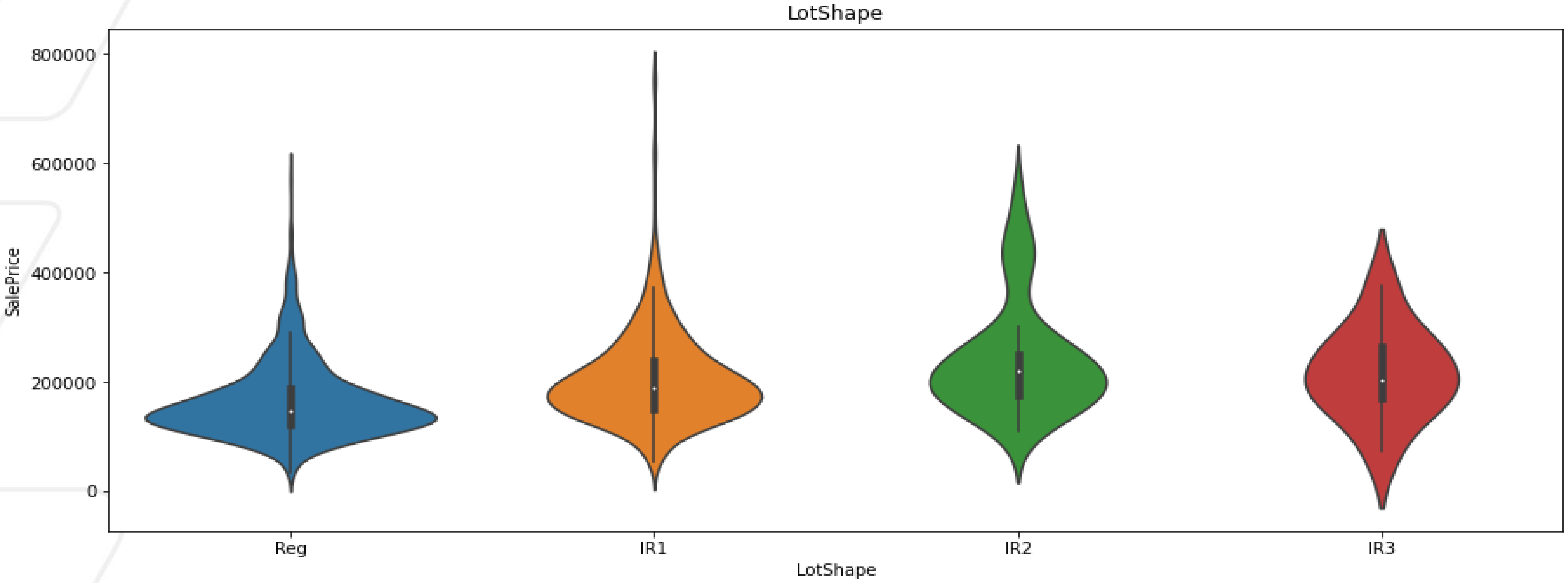
- Data Storage
  - Int64
    - Currency
    - Size
  - Object
    - Classification
- Correlated Groups
  - Bedroom vs Sales price
- Continue to Identify important features.*

Georgia Tech

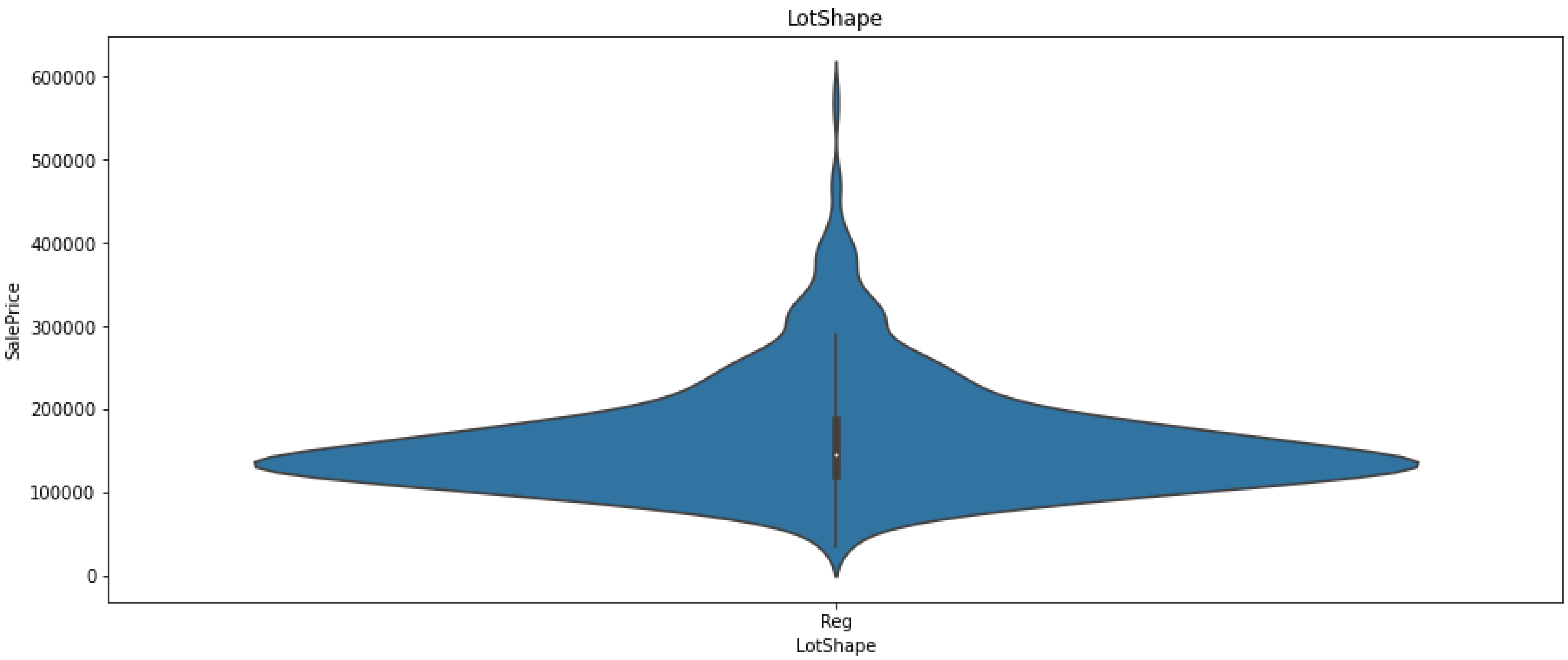
Correlation of Sale Price vs MS SubClass



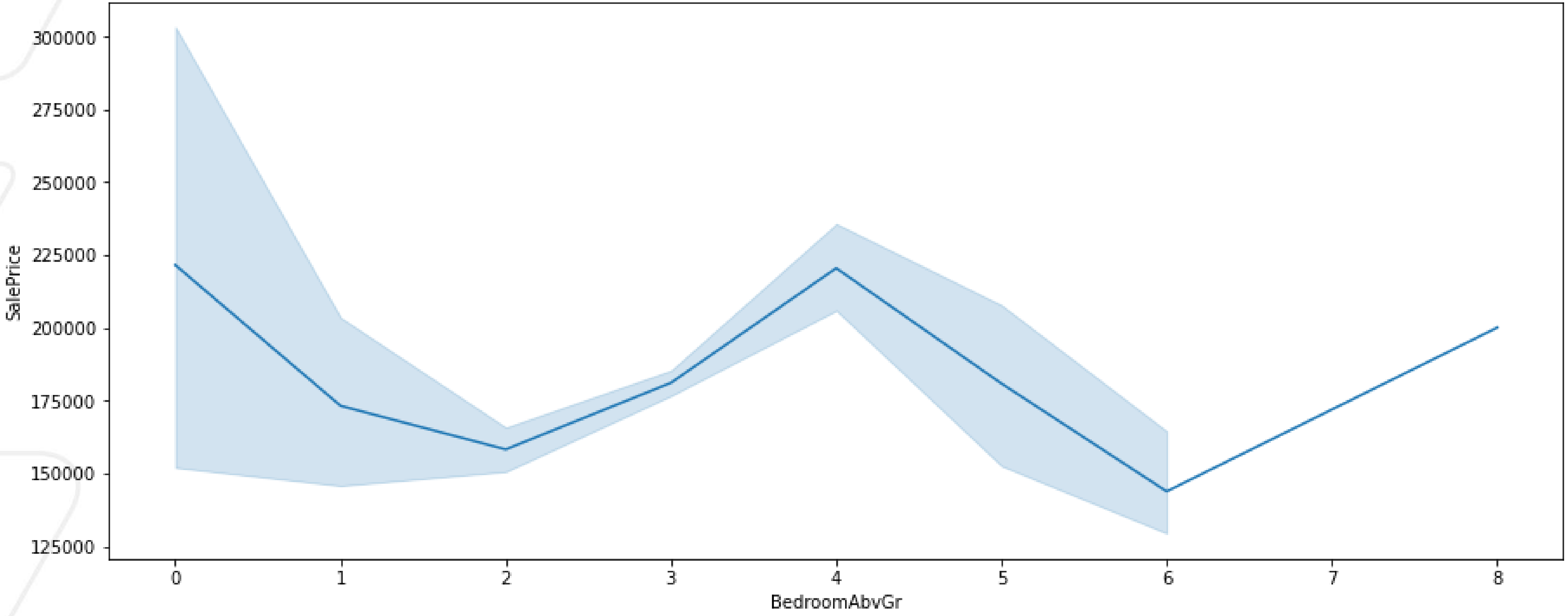
Correlation of Sale Price vs LotShape



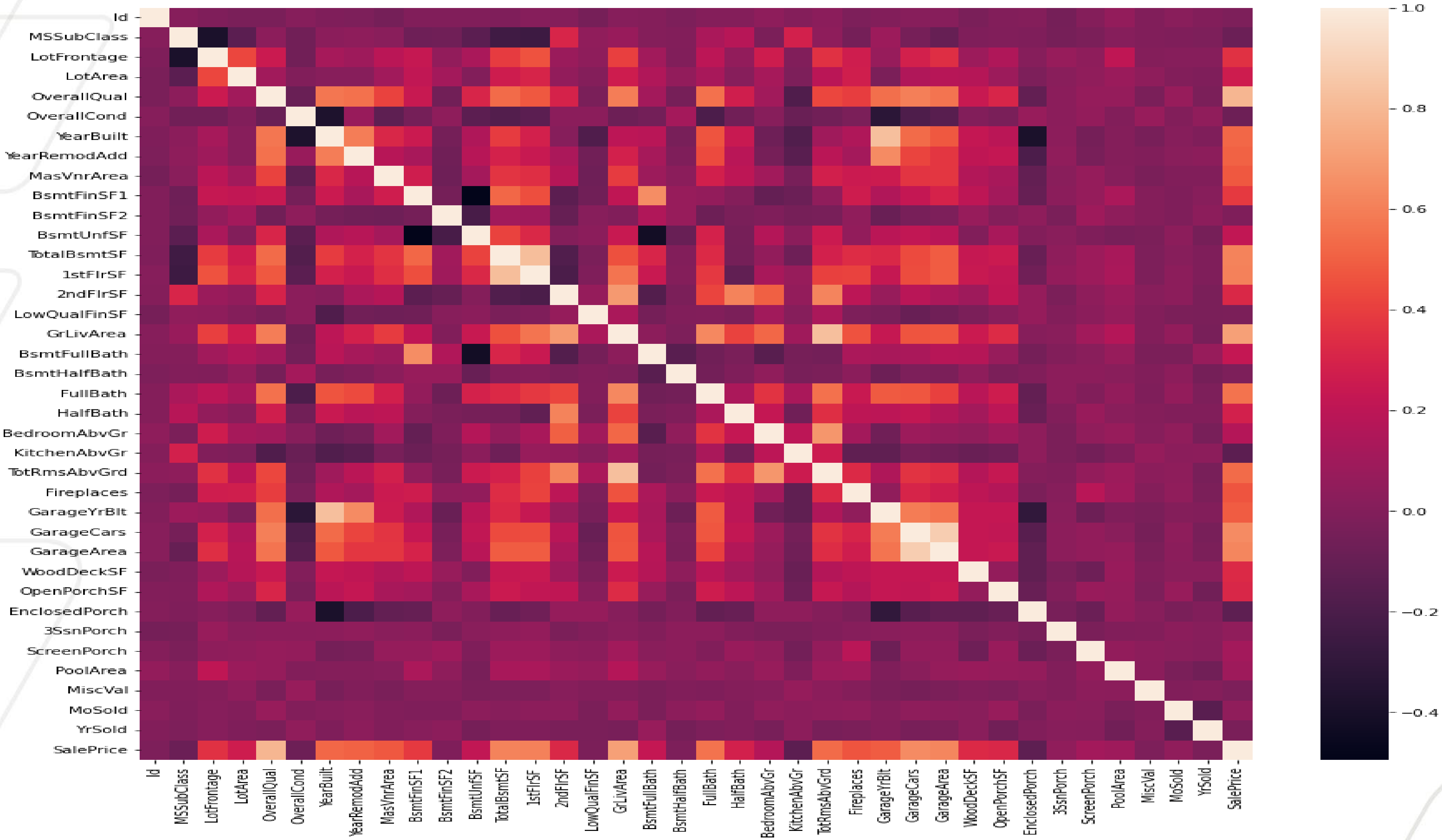
Correlation of Sale Price vs LotShape.Reg



Correlation of Sale Price vs Bedroom



Correlation Matrix

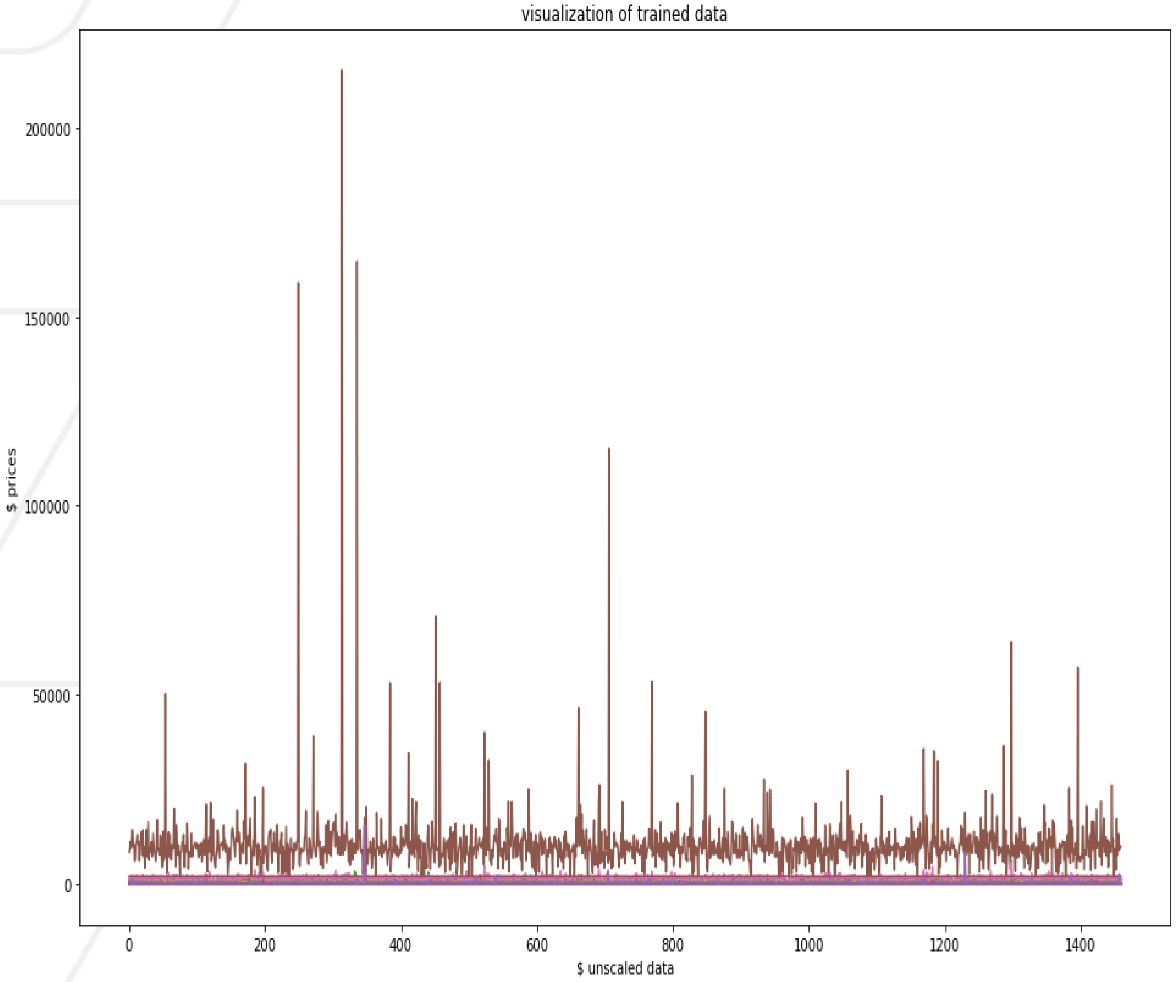


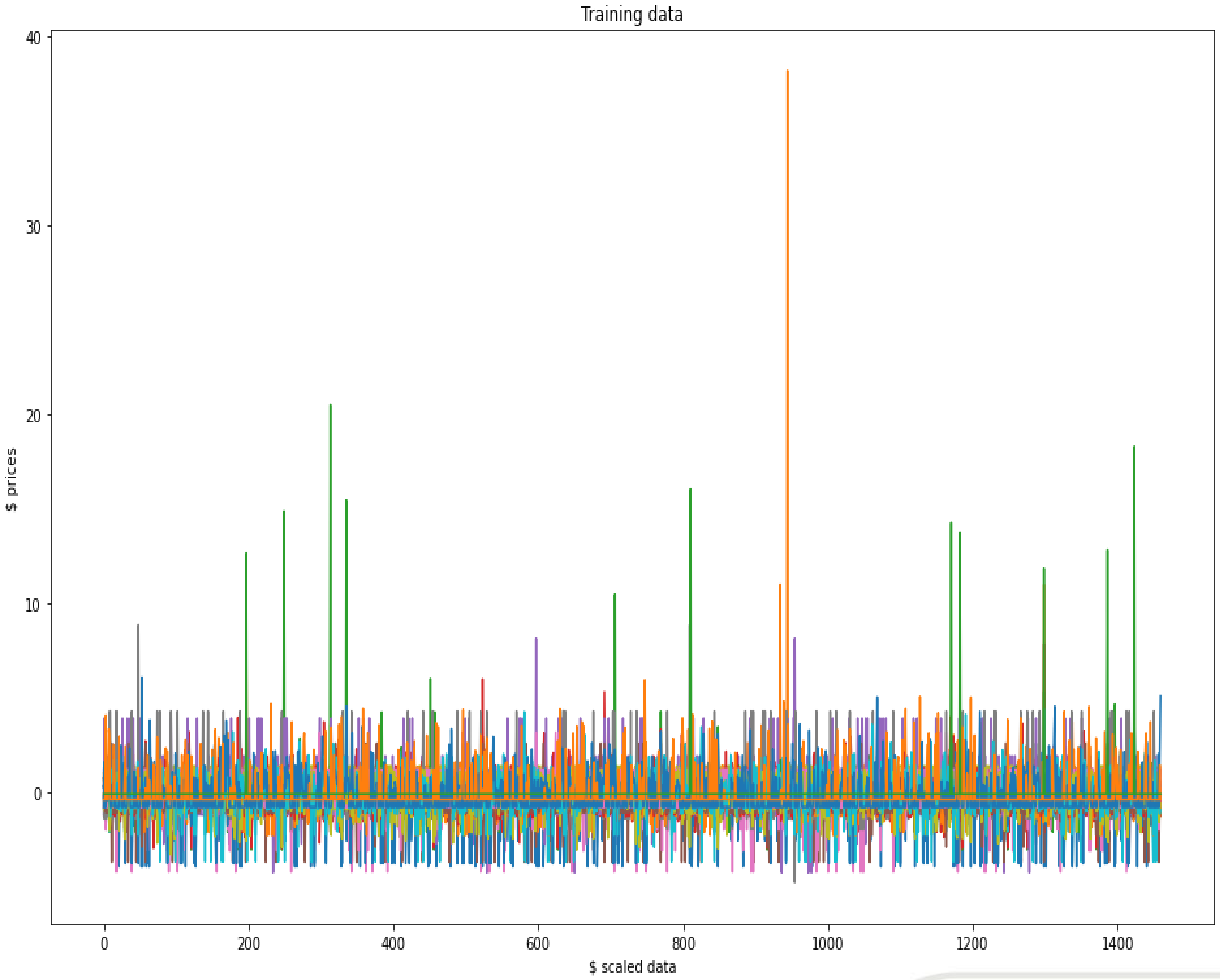
Data Processing

- Encoding Character Type
  - MS Zoning
    - Industrial, agriculture, commercial
  - Street Type
    - Paved, Gravel
- Scaling Numeric type
  - Lot Area
  - Pool Area
  - Misc Value
- Removing Pooled data
- Remove Id, Year and other non-related features.

Georgia Tech

Data Processing





Curse of Dimensionality

- Statistical models not able to compute accurate results.
- Overfitting models
- Ways to mitigate
  - Dimension reduction

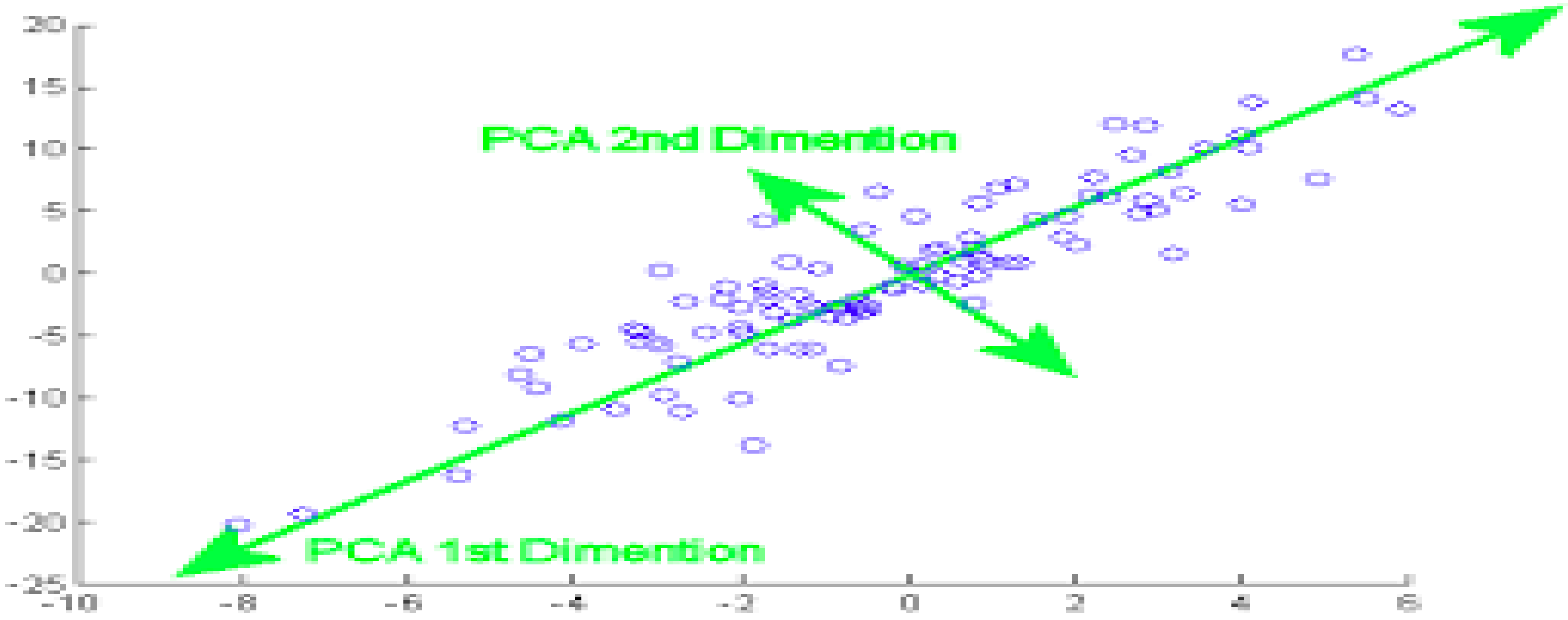
Georgia Tech

Dimension Reduction Techniques

- Principal Component Analysis(PCA)
- Functional Principal Component Analysis(FPCA)
- B-Splines
- Lasso Regression
- Removing Pooled data
- Remove Id, Year and other non-related features.

Principal Component Analysis(PCA)

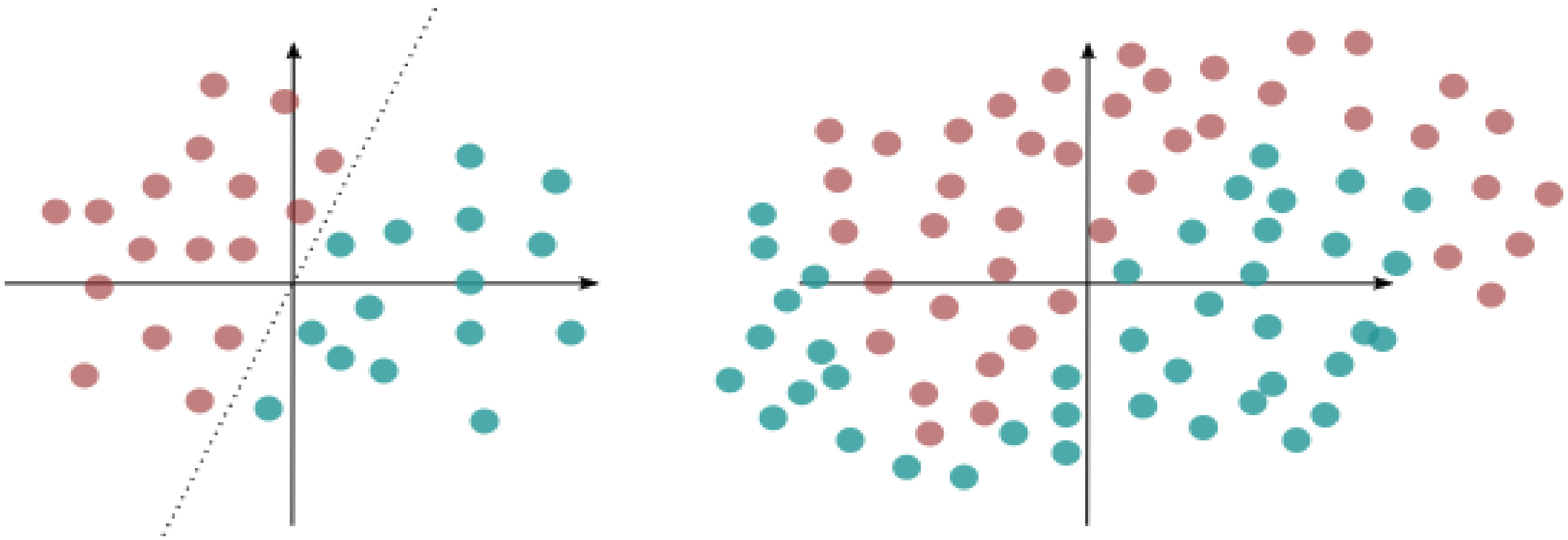
- PCA, reduction technique by weights being by variance such that majority of variance are captured by first few components. This performed by calculating eigenvalues and eigenvectors.



https://www.cs.cmu.edu/~elaw/papers/pca.pdf

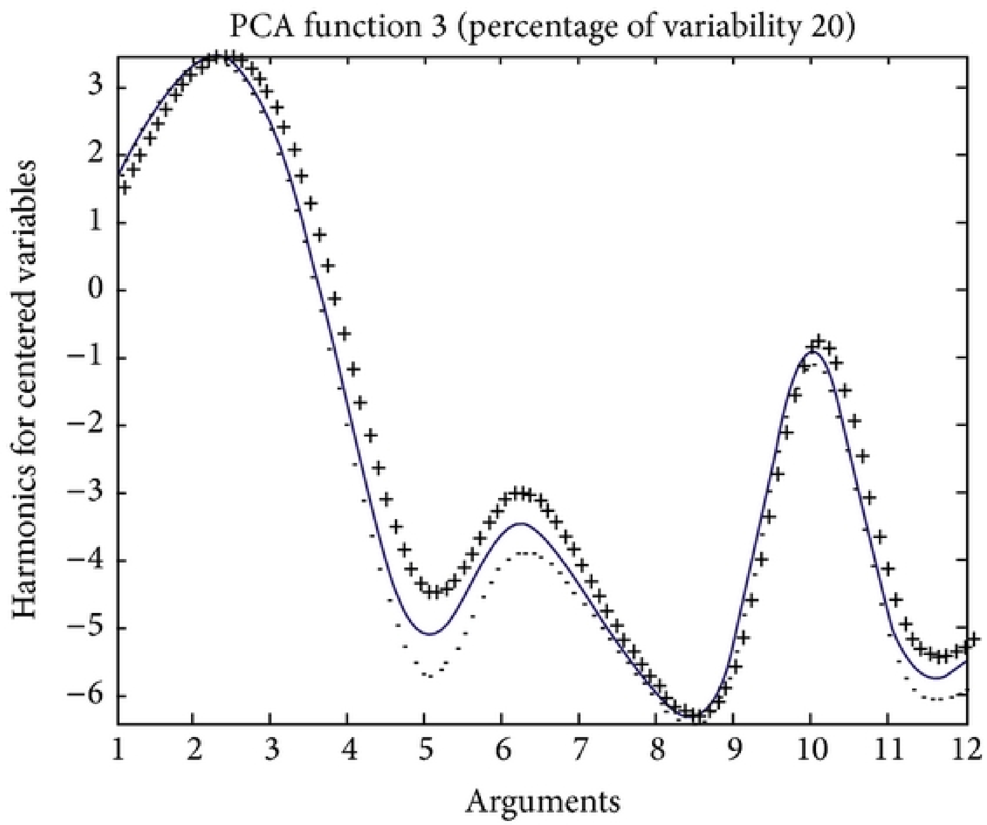
Kernel Principal Component Analysis(KPCA)

- Same as PCA with respect to eigenvalue and eigenvectors, however non-linear methods(kernel operator) is used to capture variance in low dimensional subspace



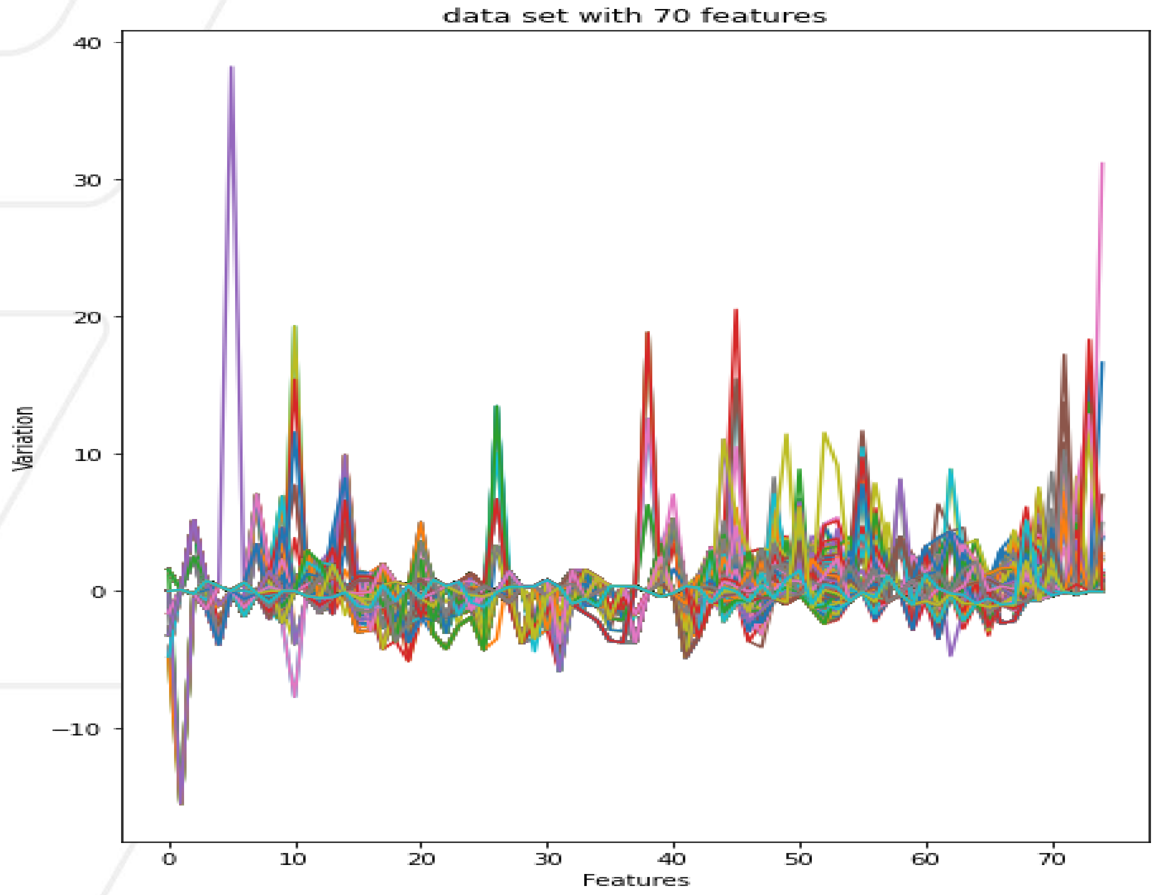
Functional PCA

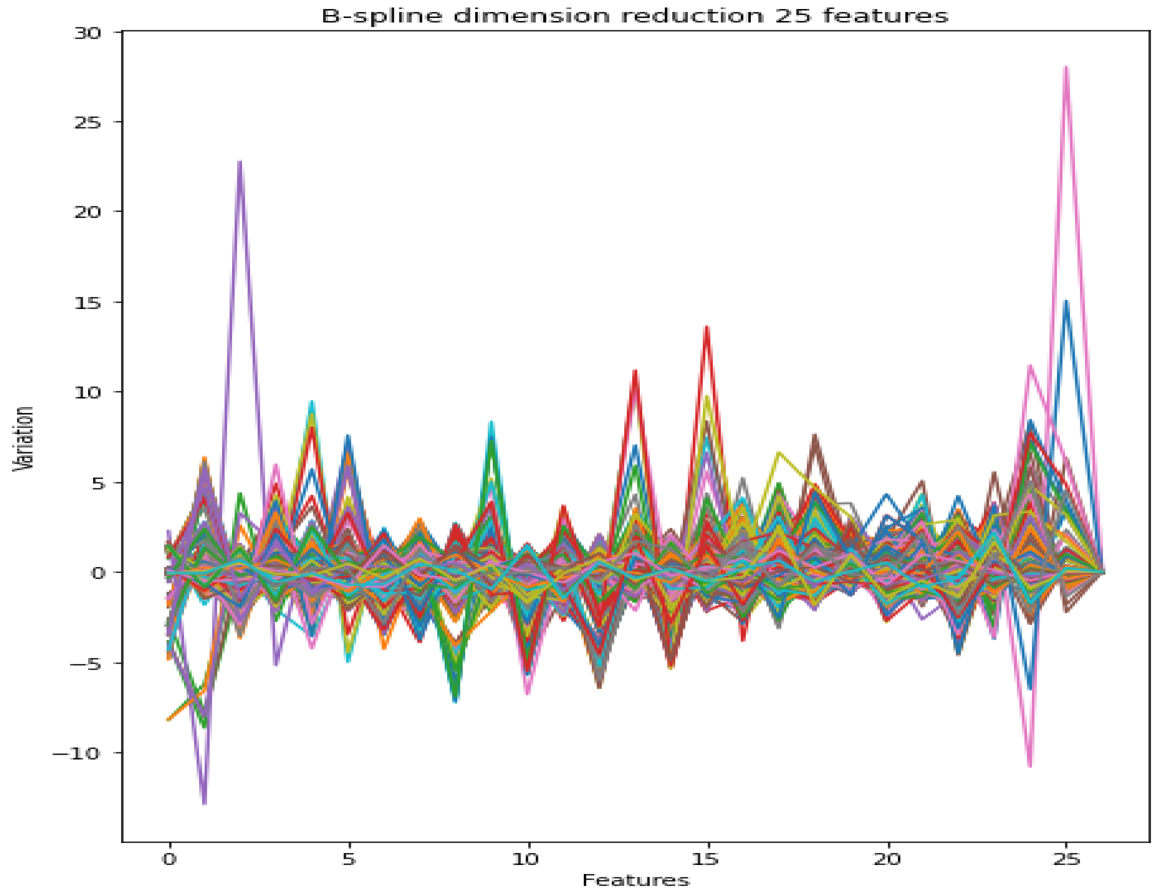
- Functional PCA, reduction technique where each point is a function input



Source: https://www.psych.mcgill.ca/misc/fda/files/CRM-FPCA.pdf

Functional Data Analysis





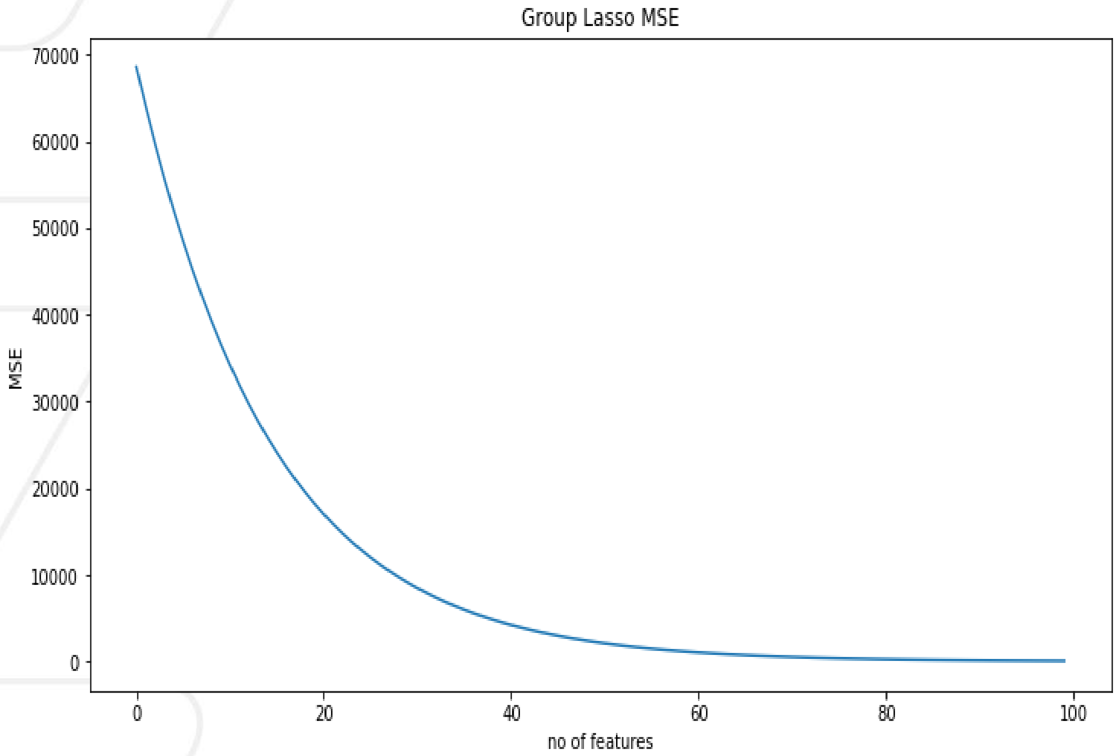
Georgia Tech

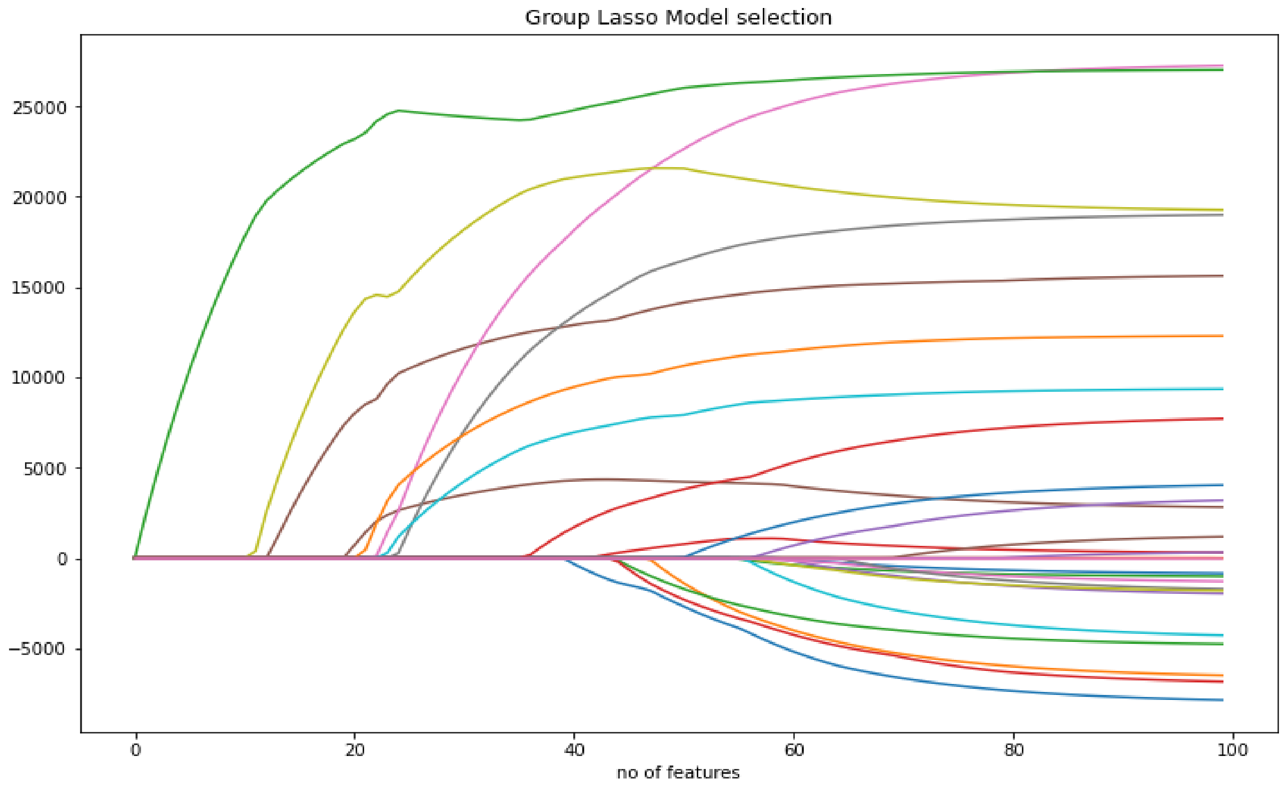
Statistical Models

- Tree based models
  - Random Forest
  - AdaBoost
  - GradientBoost
- Regression Models
  - Linear Regression
  - Ridge Regression
  - Lasso Regression
  - Adaptive Lasso Regression
  - Group Lasso

Georgia Tech

Group Lasso (MSE vs Feature selection)





OLS Statistical Sample Results

OLS Regression Results

Dep. Variable:	SalePrice	**R-squared:**	0.944
Model:	OLS	**Adj. R-squared:**	0.922
Method:	Least Squares	**F-statistic:**	44.24
Date:	Sun, 03 Jul 2022	**Prob (F-statistic):**	4.94e-324
Time:	21:28:16	**Log-Likelihood:**	-10975.
No. Observations:	978	**AIC:**	2.249e+04
Df Residuals:	709	**BIC:**	2.380e+04
Df Model:	268		
Covariance Type:	nonrobust		

	coef	std err	t	P>|t|	[0.025	0.975]
BedroomAbvGr	-2501.2766	1779.995	-1.405	0.160	-5995.969	993.416
KitchenAbvGr	-1.266e+04	9697.247	-1.305	0.192	-3.17e+04	6383.066
MSSubClass_20	-2.16e+04	1.26e+04	-1.718	0.086	-4.63e+04	3086.538
MSSubClass_30	-2.083e+04	1.28e+04	-1.628	0.104	-4.59e+04	4294.948
MSSubClass_40	-2.881e+04	2.57e+04	-1.120	0.263	-7.93e+04	2.17e+04
MSSubClass_45	-8525.9178	2.97e+04	-0.287	0.774	-6.68e+04	4.98e+04
MSSubClass_50	-1.129e+04	1.37e+04	-0.822	0.411	-3.83e+04	1.57e+04
MSSubClass_60	-1.026e+04	1.37e+04	-0.749	0.454	-3.72e+04	1.66e+04
MSSubClass_70	-3385.1479	1.31e+04	-0.258	0.797	-2.92e+04	2.24e+04

Georgia Tech

Results using Tree based models W/80-20 training/test split

Model	R^2	Mean Square Error
RandomForestRegressor	85.67%	653.289
AdaBoostRegressor	80.44%	696.289
GradientBoostRegressor	87.55%	653.289
RandomForestRegressor W/PCA	85.10%	693.289
AdaBoostRegressor W/PCA	81.53%	653.289
GradientBoostRegressor W/PCA	84.95%	693.289
RandomForestRegressor W/KPCA	78.21%	653.289
AdaBoostRegressor W/KPCA	72.06%	697.289
GradientBoostRegressor W/KPCA	78.54%	695.234

Georgia Tech

Results using Regression W/80-20 training/test split

Model	R^2	Mean Square Error
BSpines reduction with Linear Regression	78.43%	593.289
BSpines reduction with Group Lasso	76.60%	596.880
Lasso Regression with scaled data	76.88%	596.882
Ridge Regression with scaled data	74.01%	602.840
Linear Regression with scaled and w/PCA	72.05%	602.334
Linear Regression with various reduced parameters.*	88.78%	468.661

Georgia Tech

Feature Engineering

- Character column cannot be used for Machine learning. So we need to convert them to Ordinal or Label setting

```
#Nominal A variable that has no numerical importance, for example color or city.
categorical_columns_labels = ["MSSubClass", "MSZoning", "Street", "LotShape", "LandContour", "Utilities",
                               "LotConfig",  "LandSlope", "Neighborhood", "Condition1", "Condition2", "BldgType",
                              "HouseStyle", "RoofStyle", "RoofMatl", "Exterior1st", "Exterior2nd",
                             "Foundation", "Heating","CentralAir","Functional","PavedDrive","SaleType","SaleCondition"]

#Ordinal A variable that has some order associated with it like our place example above
categorical_columns_ranking = ["OverallQual", "OverallCond", "ExterQual", "ExterCond", "BsmtQual", "BsmtCond",
                              "BsmtExposure","HeatingQC","KitchenQual","FireplaceQu","GarageQual","GarageCond",
                              "PoolQC","Fence"]

#Nominal A variable that has no numerical importance, for example color or city.
categorical_columns_ordinal = ["BsmtFullBath", "BsmtHalfBath", "FullBath",
                              "HalfBath", "BedroomAbvGr","KitchenAbvGr","TotRmsAbvGrd", "Fireplaces",
                              "GarageCars","BsmtFinSF1","BsmtFinSF2", "BsmtUnfSF", "TotalBsmtSF",
                             "1stFlrSF","2ndFlrSF", "LowQualFinSF","BsmtFullBath"]

categorical_columns_year = ["YearBuilt", "YearRemodAdd", "MoSold","YrSold"]

continious_columns = ["LotArea","GrLivArea", "GarageArea","WoodDeckSF","OpenPorchSF",
                      "EnclosedPorch","3SsnPorch", "ScreenPorch", "PoolArea"]

continous_currency = ["MiscVal"]
```

Georgia Tech

Feature Selection using Lasso and Extra tree classifier

```
from sklearn.linear_model import LassoCV

reg = LassoCV(cv=5, random_state=42, fit_intercept=False).fit(X_train,y_train)
X_train.columns[reg.coef_>= 1e-9]
```

```
Index(['Neighborhood', 'BsmtFinSF1', 'BsmtFinSF2', 'BsmtUnfSF', 'TotalBsmtSF',
       '1stFlrSF', '2ndFlrSF', 'ScreenPorch'],
      dtype='object')
```

```
from sklearn.ensemble import ExtraTreesClassifier
clf = ExtraTreesClassifier(n_estimators=50)
clf = clf.fit(X_train, y_train)
min_val = np.min(clf.feature_importances_[clf.feature_importances_ < 5])
max_val = np.max(clf.feature_importances_)
X_train.columns[np.argwhere(clf.feature_importances_ > 0).reshape(-1)]
```

```
Index(['MSSubClass', 'MSZoning', 'Street', 'LotShape', 'LandContour',
       'Utilities', 'LotConfig', 'LandSlope', 'Neighborhood', 'Condition1',
       'Condition2', 'BldgType', 'HouseStyle', 'RoofStyle', 'RoofMatl',
       'Exterior1st', 'Exterior2nd', 'Foundation', 'Heating', 'CentralAir',
       'Functional', 'PavedDrive', 'SaleType', 'SaleCondition', 'BsmtFullBath',
       'BsmtHalfBath', 'FullBath', 'HalfBath', 'BedroomAbvGr', 'KitchenAbvGr',
       'TotRmsAbvGrd', 'Fireplaces', 'GarageCars', 'BsmtFinSF1', 'BsmtFinSF2',
       'BsmtUnfSF', 'TotalBsmtSF', '1stFlrSF', '2ndFlrSF', 'LowQualFinSF',
       'OverallQual', 'OverallCond', 'ExterQual', 'ExterCond', 'BsmtQual',
       'BsmtCond', 'BsmtExposure', 'HeatingQC', 'KitchenQual', 'FireplaceQu',
       'GarageQual', 'GarageCond', 'PoolQC', 'Fence', 'LotArea', 'GrLivArea',
       'GarageArea', 'WoodDeckSF', 'OpenPorchSF', 'EnclosedPorch', '3SsnPorch',
       'ScreenPorch', 'PoolArea'],
      dtype='object')
```

Results using Tree based models W/80-20 training/test split

Model	R^2	Mean Square Error
RandomForestRegressor W/Feature Engineering & Feature Selection	89.92%	653.289
AdaBoostRegressor W/Feature Engineering & Feature Selection	88.82%	696.289
GradientBoostRegressor W/Feature Engineering & Feature Selection	91.44%	653.289
RandomForestRegressor W/PCA	85.10%	693.289
AdaBoostRegressor W/PCA	81.53%	653.289
GradientBoostRegressor W/PCA	84.95%	693.289
RandomForestRegressor W/KPCA	78.21%	653.289
AdaBoostRegressor W/KPCA	72.06%	697.289
GradientBoostRegressor W/KPCA	78.54%	695.234

Summary

- TO DO
- What’s Next