# DSI Project 2

Ames Housing Data and Kaggle Challenge

Alex
Darion
Richa
Zaini

# Content

- Problem Statement

- Exploratory Data Analysis

- Feature Engineering

- Model Building
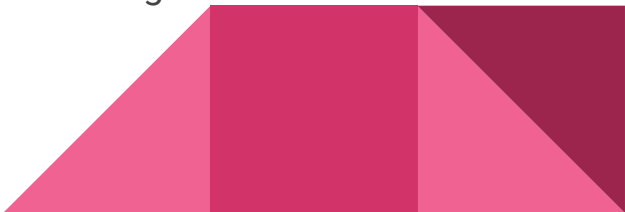
- Model Evaluation

- Conclusions & Recommendations

# Problem Statement

In this project, we aim to **create and iteratively refine a regression model to predict the price** of a property in Ames, Iowa. This can help to prevent over or under selling, and assist **both prospective buyers and sellers** to reach their goals more effectively.

We will like to better understand which features of a property will potentially increase the sale price of a house, as well as what features may hurt the prospects of a house sale. This information is extremely valuable to **prospective sellers, as they will then be able to carry out targeted improvements** to their home before trying to sell it.

This model can also be **useful to developers** who are looking to develop an area in a certain neighborhood. They will be able to negotiate at a better price with each other during a transaction and achieve a win-win situation.

# Available Data

1) We are given 2 datasets, 1 each for training and for testing
   a) The train dataset has 2051 rows, each representing a housing sale, and 81 columns representing different features of the property
   b) The test dataset has 879 rows and 80 columns (minus SalePrice which the model will help us predict)
2) Features include both numeric (lot frontage, lot area) and and categorical variables (sale type, overall quality, overall condition)

# Exploratory Data Analysis

1. Inspecting Data

2. Handling Missing Data

3. Refining Data Types

4. Reviewing of Features

5. Exploratory Visualisations

    a. Distribution of features

    b. Correlation of features

# 1. Inspecting Data

Potential Outliers
- We compared the ground living area and the sale price. Potential outliers were identified. Apparently, two properties have a very low price for a very large ground living area. So we decided to remove these outliers.



Distribution of SalePrice



Gr Liv Area against SalePrice
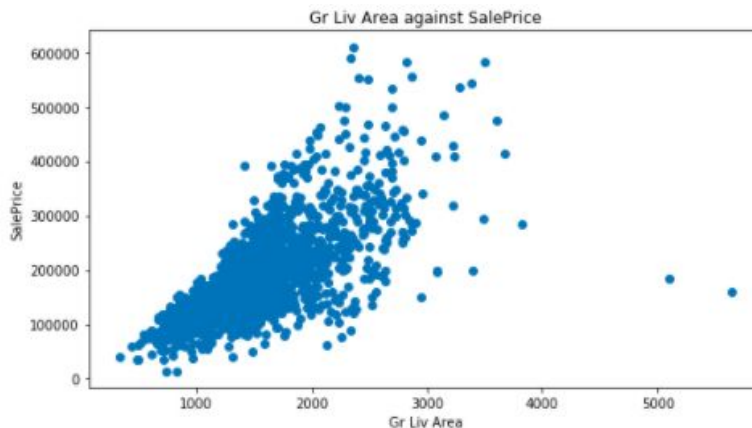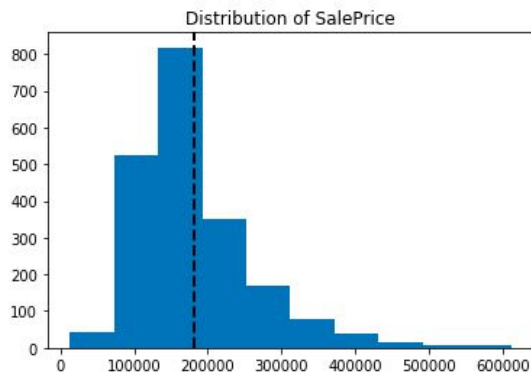
| | saleprice | gr liv area |
|---|---|---|
| 616 | 284700 | 3820 |
| 960 | 160000 | 5642 |
| 1885 | 183850 | 5095 |

# 2. Data Cleaning - Null Values

- The training data came with an abundance of null values.
- For some of the missing nulls we made educated guesses, for examples, entries under Garage-'suffix' were null across the board, which suggested the lack of a garage.
- Larger values of missing nulls indicate more intentionality and would also suggest the lack of- rather than a simple error.
- Using other pieces of data contextually, we could make best guesses for some of the remaining missing values.

| | |
|---|---|
| Pool QC | 2042 |
| Misc Feature | 1986 |
| Alley | 1911 |
| Fence | 1651 |
| Fireplace Qu | 1000 |
| Lot Frontage | 330 |
| Garage Qual | 114 |
| Garage Finish | 114 |
| Garage Cond | 114 |
| Garage Yr Blt | 114 |
| Garage Type | 113 |

| | |
|---|---|
| Bsmt Exposure | 58 |
| BsmtFin Type 2 | 56 |
| Bsmt Cond | 55 |
| Bsmt Qual | 55 |
| BsmtFin Type 1 | 55 |
| Mas Vnr Type | 22 |
| Mas Vnr Area | 22 |
| Bsmt Half Bath | 2 |
| Bsmt Full Bath | 2 |
| Garage Cars | 1 |
| Garage Area | 1 |
| Total Bsmt SF | 1 |
| Bsmt Unf SF | 1 |
| BsmtFin SF 2 | 1 |

# 2. Imputing Missing Data - Garage Year Built?

- Data on Garages seems to show that missing data is due to absence of feature
- There are multiple ways to handle the data for 'Year Built'

```python
def Garage_Age(date):

    if date == 'NA':
        age = 0

    elif date < 1950:
        age = 1

    elif date < 1960:
        age = 2

    elif date < 1970:
        age = 3

    elif date < 1980:
        age = 4

    elif date < 1990:
        age = 5

    elif date < 2000:
        age = 6

    else:
        age = 7

    return age
```
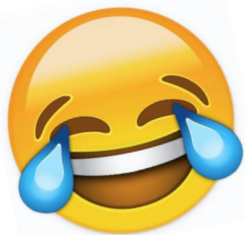
# 3. Refining Data Types

There are three type of data in the data sets:

1) Nominal
2) Ordinal
3) Continuous

Methods used to refine:

1) One Hot Encoding for Nominal Data
2) Binarizing Ordinal Data

```python
#Apply dummies to the below features
train = pd.get_dummies(train, columns=['ms subclass','ms zoning',
                                       'lot config','neighborhood','condition 1',
                                       'bldg type','house style',
                                       'roof style','exterior 1st',
                                       'exterior 2nd','foundation',
                                       'sale type','lot shape',
                                       'land contour','mas vnr type',
                                       'garage type'],drop_first=True)
```

```python
housing.replace(to_replace = {
    'Bsmt Cond': {'Ex': 5, 'Gd': 4, 'TA':3, 'Fa':2, 'Po': 1, 'NA': 0},
    'Bsmt Exposure': {'Gd': 4, 'Av':3, 'Mn':2, 'No': 1, 'NA': 0},
    'Bsmt Qual': {'Ex': 5, 'Gd': 4, 'TA':3, 'Fa':2, 'Po': 1, 'NA': 0},
    'BsmtFin Type 1': {'GLQ': 6, 'ALQ': 5, 'BLQ': 4, 'Rec': 3, 'LwQ': 2, 'Unf': 1, 'NA': 0},
    'BsmtFin Type 2': {'GLQ': 6, 'ALQ': 5, 'BLQ': 4, 'Rec': 3, 'LwQ': 2, 'Unf': 1, 'NA': 0},
    'Electrical': {'SBrkr': 4, 'FuseA': 3, 'FuseF':2, 'FuseP':1, 'Mix': 0},
    'Exter Cond': {'Ex': 4, 'Gd': 3, 'TA':2, 'Fa':1, 'Po': 0},
    'Exter Qual': {'Ex': 4, 'Gd': 3, 'TA':2, 'Fa':1, 'Po': 0},
```

# 4. Reviewing of Features

- Distribution
- Correlation

| | 1st Flr SF | 2nd Flr SF | Low Qual Fin SF | Gr Liv Area | SalePrice |
|---|---|---|---|---|---|
| **1st Flr SF** | 1.000000 | -0.284643 | -0.025900 | 0.524200 | 0.655012 |
| **2nd Flr SF** | -0.284643 | 1.000000 | -0.011392 | 0.661448 | 0.251401 |
| **Low Qual Fin SF** | -0.025900 | -0.011392 | 1.000000 | 0.070367 | -0.047147 |
| **Gr Liv Area** | 0.524200 | 0.661448 | 0.070367 | 1.000000 | 0.727456 |
| **SalePrice** | 0.655012 | 0.251401 | -0.047147 | 0.727456 | 1.000000 |
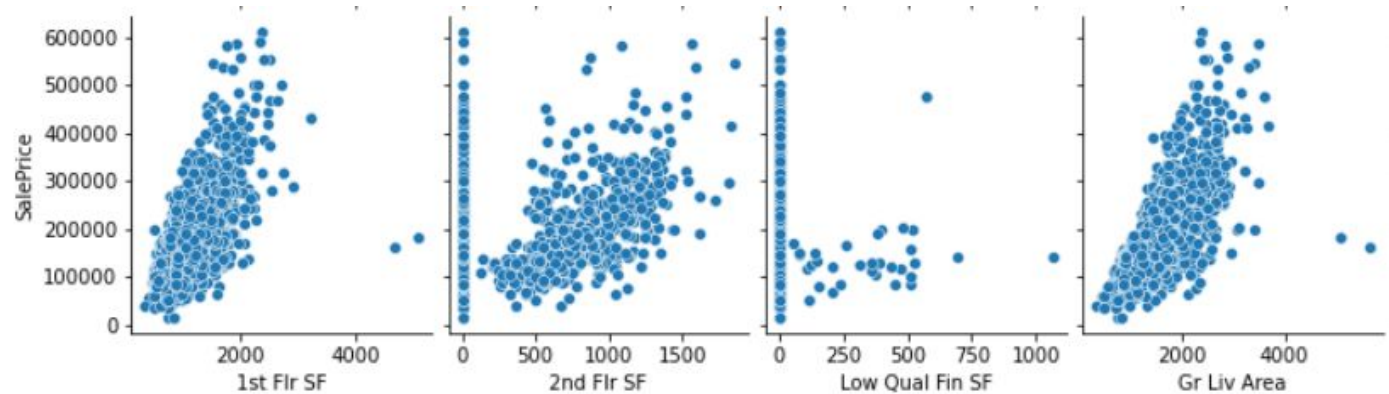
# Feature Engineering - Total living Area and 2nd Flr Area

- There is a strong correlation between total living area and 2nd floor area

- We created an interaction term between the two of them by multiplying the columns together

|  | 1st Flr SF | 2nd Flr SF |
|---|---|---|
| 1st Flr SF | 1.000000 | -0.284643 |
| 2nd Flr SF | -0.284643 | 1.000000 |
| Low Qual Fin SF | -0.025900 | -0.011392 |
| Gr Liv Area | 0.524200 | 0.661448 |
| SalePrice | 0.655012 | 0.251401 |



Gr Liv Area

2nd Flr SF
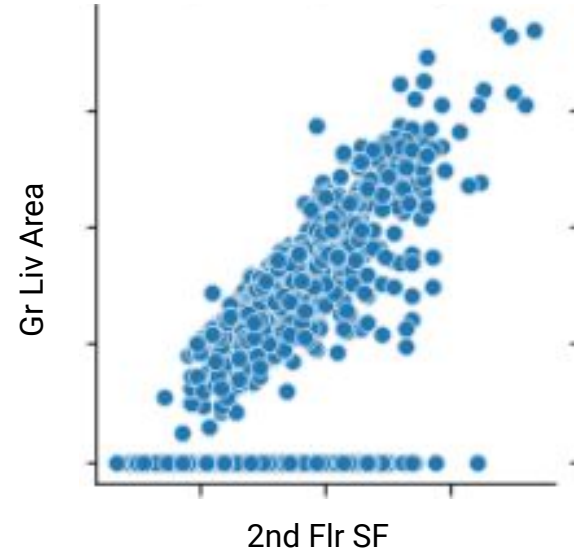
# Feature Engineering - Total Basement SF

- We have grouped some related
  features together to find the
  multicollinearity between them.
- An example will be total
  basement square feet vs
  basement finished SF 1,
  basement finish SF 2 &
  basement unfinished SF.



Total Basement Square Feet vs Basement Finished Square Feet 1, Basement Finish Square Feet 2 and Basement Unfinished Squre Feet
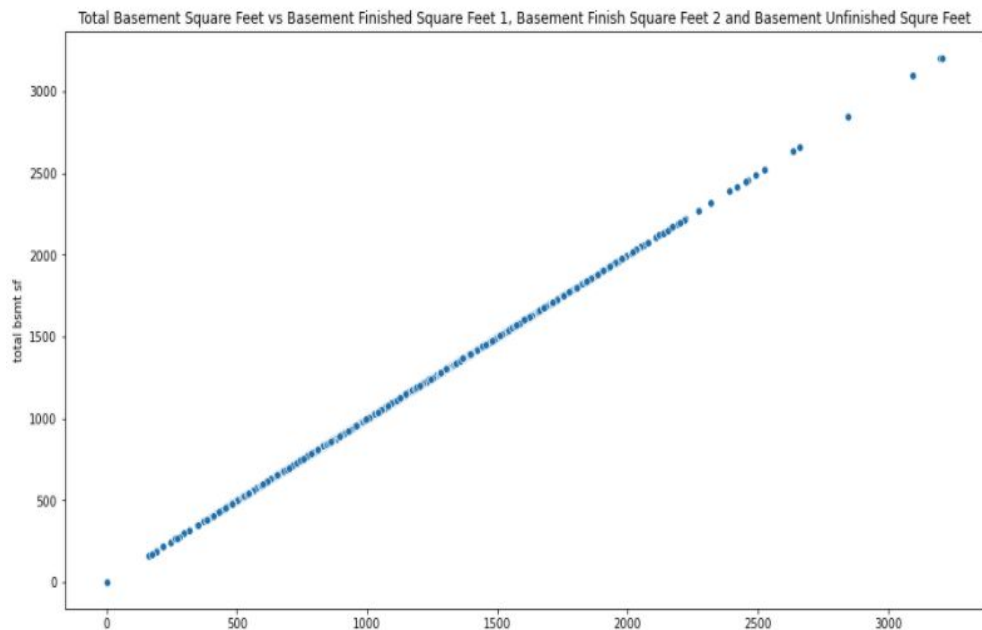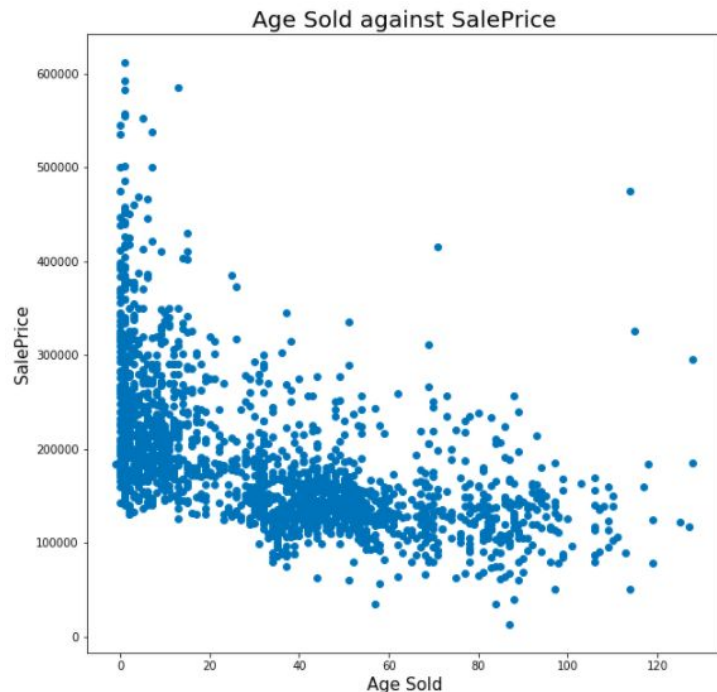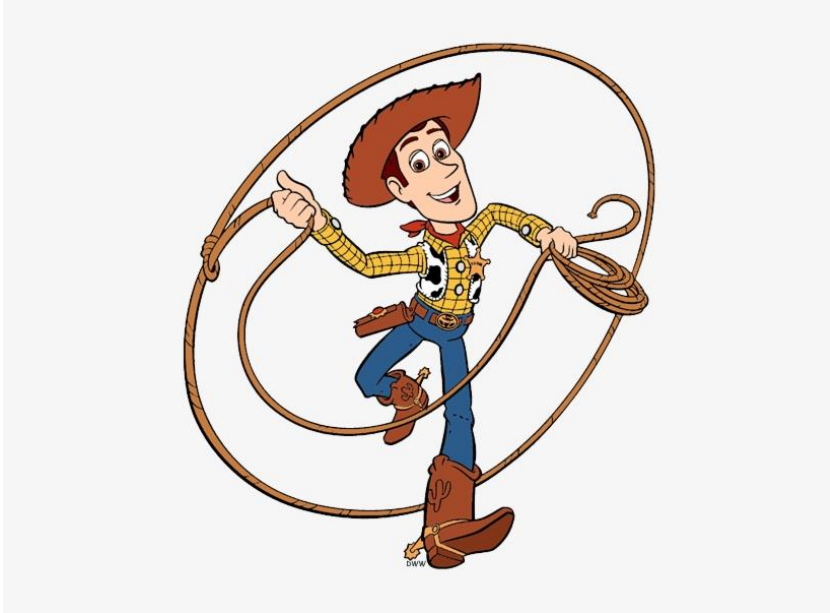
# Feature Engineering

We also used Year Built and Year Sold to compute age of the house at the time of sale

As expected, the new feature seems to be negatively correlated with Sale Price

With Age Sold, we can drop Year Built and Year Sold

# Feature Selection - Lasso

# Feature Selection - Lasso

- After carrying out some manual feature engineering through EDA, we run a LassoCV model to extract the most significant features.

- We then select ~30 features to run our final model on.

```
In [26]: lasso_coef[lasso_coef['Coef'] != 0]
Out[26]:
```

| | Coef |
|---|---|
| Gr Liv Area | 15977.143740 |
| Overall Qual | 11375.216542 |
| Total Bsmt SF | 8756.170340 |
| BsmtFin SF 1 | 8643.835651 |
| Gr * 2nd Flr SF | 6649.451064 |
| ... | ... |
| Bsmt Cond | -2114.511188 |
| Roof Style_Mansard | -2386.959796 |
| Mas Vnr Type_BrkFace | -2622.287293 |
| House Age/Remod | -3304.440763 |
| House Age | -4482.171566 |

122 rows × 1 columns

# Polynomial Featuring

- A group member tried out Polynomial Transform after selecting the top 20 significant features.
- As it turns out, overall quality and living area are great predictors to the sale price of the property.
- Second to that, space in living areas drives property prices.

| | cross_lasso_coef |
|---|---|
| overallqual grlivarea | 22332.831999 |
| overallqual totalbsmtsf | 11230.168666 |
| exterqual 1stflrsf | 9491.450044 |
| grlivarea kitchenqual | 9409.077022 |
| bsmtfinsf1 bsmtqual | 8398.212179 |
| bsmtqual lotarea | 7893.851745 |
| exterqual bsmtfinsf1 | 7474.638404 |
| garagecars masvnrarea | 7149.667280 |
| yearbuilt | 6955.356019 |
| neighborhood_nridght masvnrarea | 5850.418403 |
| neighborhood_stonebr saletype_new | 5526.362356 |
| bsmtexposure masvnrarea | 5349.586740 |
| overallqual lotarea | 5229.672350 |
| neighborhood_stonebr lotarea | 5213.551578 |
| neighborhood_nridght saletype_new | 5180.371860 |
| bsmtfinsf1 yearbuilt | 4331.471078 |
| bsmtqual totalbsmtsf | 4157.320869 |
| kitchenqual lotarea | 3878.979819 |
| kitchenqual 1stflrsf | 3874.718972 |
| grlivarea exterqual | 3367.114289 |

# Polynomial Features

| | |
|---|---|
| 1stflrsf totalbsmtsf | -3050.555350 |
| masvnrarea lotarea | -4584.422594 |
| bsmtfinsf1 1stflrsf | -4588.793175 |
| grlivarea bsmtfinsf1 | -4968.495143 |

Overfitting??

- While this step did seem to improve the predicted values, it resulted in some perplexing coefficient values

- Prediction vs Explainability tradeoff
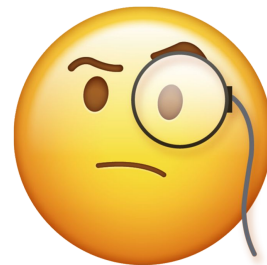
# Model Building

1) Linear Regression Model
2) Lasso Regression Model
3) Ridge Regression Model

All of us attempted the model building using the above methods. Out of the three, all of us went ahead with Lasso Regression Model as it came out with the best R2 and RMSE.

# Model Evaluation - Baseline Model

- We create a baseline model that will carry out predictions **before** any feature engineering is done

- This allows us to check if any feature engineering done will improve our predictions or make it worse

| Submission and Description | Private Score | Public Score |
|---|---|---|
| Projected Prices.csv<br>41 minutes ago by Ahmad Zaini Chia<br>add submission details | 30664.76719 | 27971.22437 |
| Projected Prices (Baseline).csv<br>41 minutes ago by Ahmad Zaini Chia | 31355.10280 | 34341.94705 |

# Model Evaluation - Residual Plots

- We carry out predictions on the training data and compare to the actual sale prices

- The house prices between $100 000 and $300 000 are reasonably well predicted

- Beyond that range the predictions are not as accurate
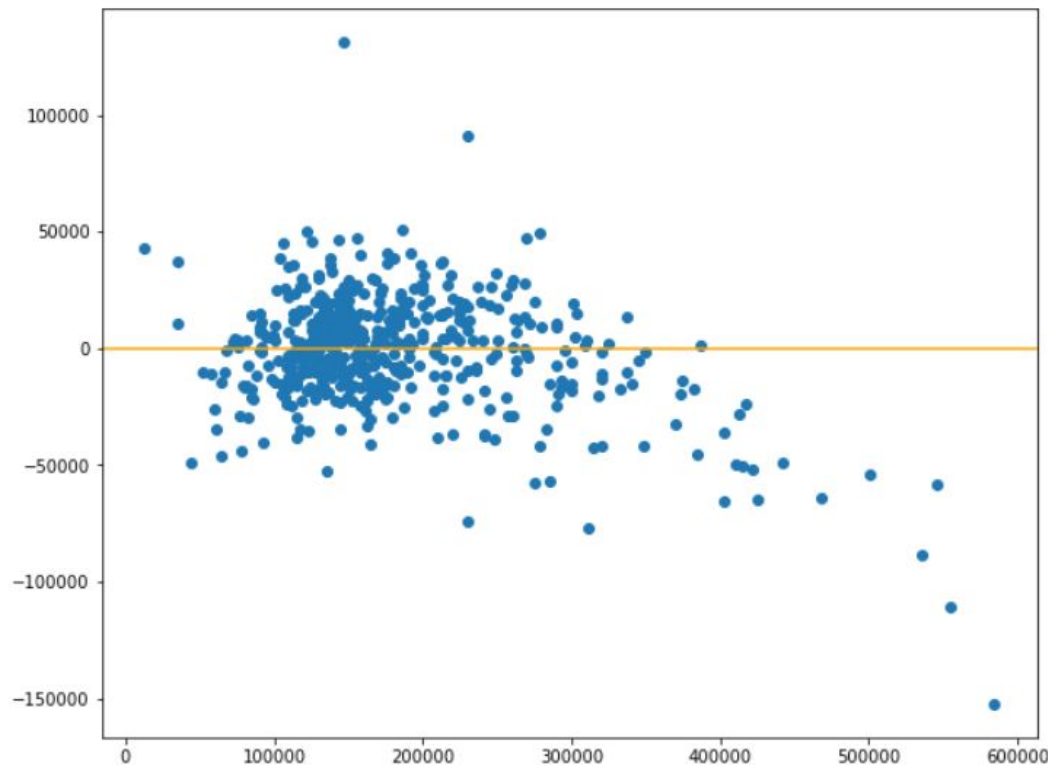
# Conclusions

Key Takeaways from the Project:

a) There were alot of null values. We have cleaned them up by understanding the reason behind the null values.
b) Certain features are correlated. Therefore, feature engineering is used to sort the features out and  pick which features to use.
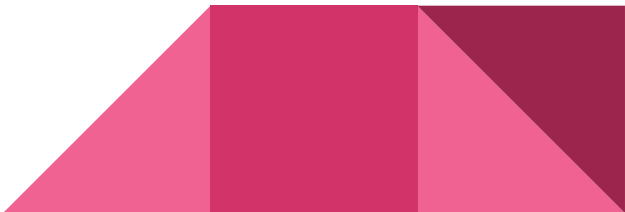
# Recommendations - Features

1. Living area above ground (including 1st and 2nd floor area)
2. Overall material and finish of the house, as well as overall condition and functionality
3. Basement area, how much of it is finished properly, and exposure
4. Contour of land and lot area
5. Age of Garage and how many cars it can fit
6. Exterior quality, Masonry veneer area
7. Kitchen quality
8. Number of fireplaces

# Recommendations - Neighbourhoods

Our data shows that the neighborhoods that are preferred by buyers are:

1. Northridge Heights
2. Stone Brook
3. Northridge
4. Crawford

There are a few neighborhoods which buyers seem to less keen on:

1. North Ames
2. Old Town
3. Edwards
4. College Creek

# Improvements

1) More features can included such as nearest public transportation, how many times the property had been switched hands and etc...

2) Additionally, the focus of the project could be switched, since the best predictors tend to be obvious even to a layperson, perhaps studying the 5th-10th best predictors could give developers the means to edge out the competition.
   - E.g Adding a fireplace or a specific roof tile would drive property prices

# Is the model generalizable?

- In order for the model to be used widely, the model would need to focus on the more general features such as area, number of rooms that will be readily available across cities.

- Conversely, we will need to go through the exploratory data analysis again to understand the nuances of housing in the particular area and adapt the model accordingly. However, this will require greater resource investment.