NANYANG TECHNOLOGICAL UNIVERSITY SINGAPORE

Exploratory Data Analysis (EDA) summary

Smitha K G
Senior Lecturer
College of Computing and Data Science

Learning objective and Overview

- Understand the role of EDA
- Identify data quality and distribution issues
- Apply cleaning, encoding, transformation, and scaling
- Evaluate model performance before and after EDA

Dataset Overview

- Kaggle: House Prices – Advanced Regression Techniques
- 1,460 observations, 80+ features
- Mixed data types: numerical, ordinal, nominal
- Target variable: SalePrice

Steps of EDA

Data Inspection/exploration

- df.shape → dataset scale
- df.info() → data types & missing values
- Early detection of potential issues
- No transformations yet

Statistical Summary

- Mean, median, min, max, std (univariate data and viz.)
- Large gaps between mean and median indicate skew
- Wide ranges indicate scaling issues
- Early hint of outliers

Steps of EDA- continued

Data Cleaning

- Correcting invalid values
- Resolving inconsistencies (in naming)
- Enforcing semantic correctness
- Making data model-consumable

Converting *Ambiguous NaNs* into Meaningful Categories

In this dataset, many NaNs do NOT mean “unknown”.

Ex: Garage related features like GarageType, GarageFinish, GarageQual,GarageCond, GarageCars, GarageArea (if the have Nan, we have to put that as None (categorical) or 0(numerical) as this only means no Garage)

Missing is the absence of structure

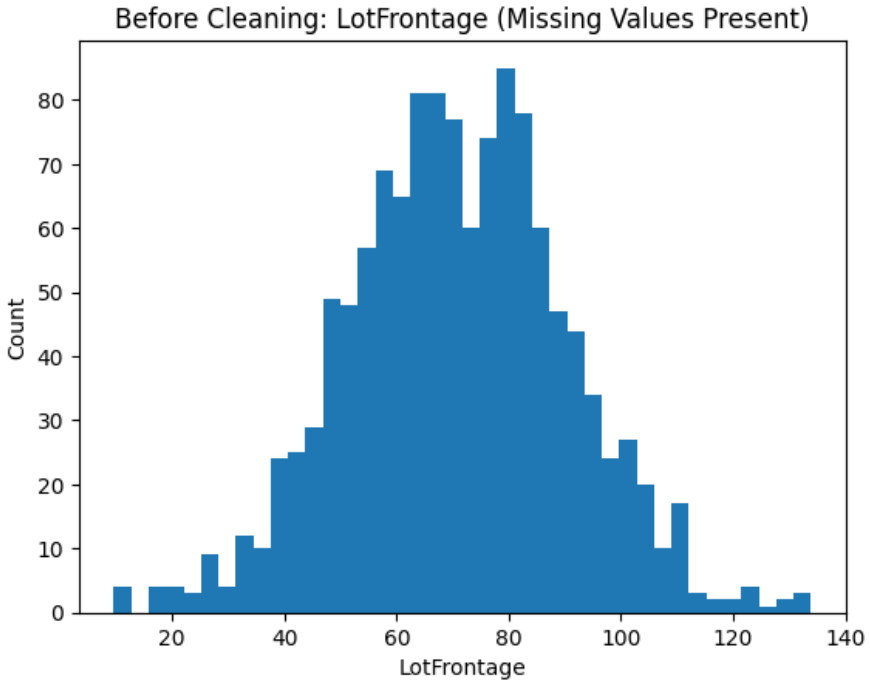
Prevents misinterpretation as bad data and Preserves domain meaning

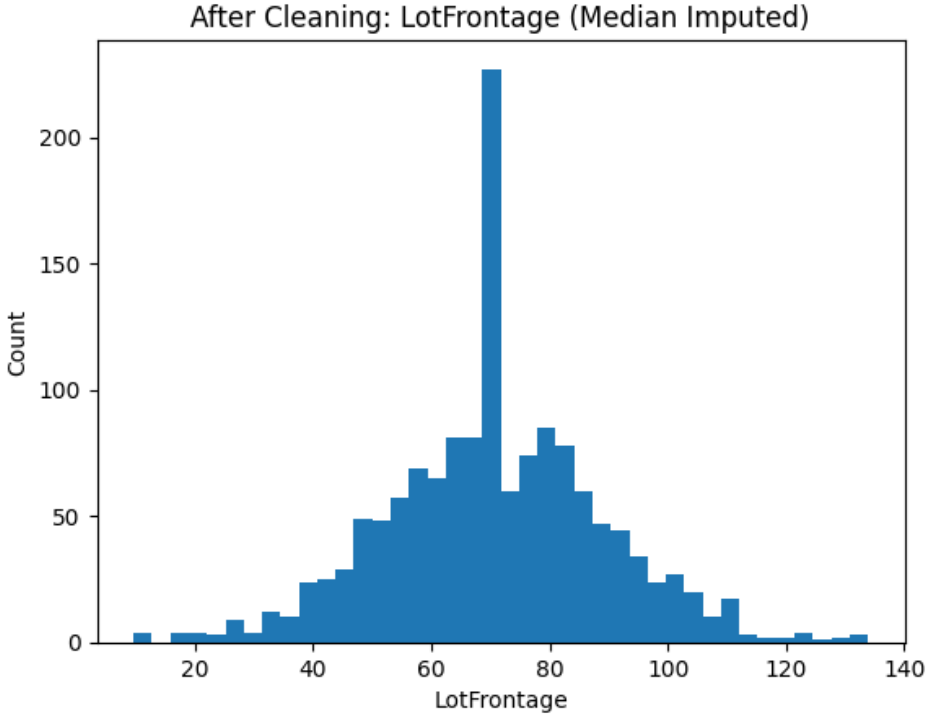
Steps of EDA- continued

Missing Value Analysis

- Not all missing values mean data error
- Different features require different strategies
- Missingness itself carries information
- Features like LotFrontage, MasVnrArea, Electrical, KitchenQual if missing need to be imputed- (MCAR, MAR, MNAR)

Feature	Why Missing is OK	Correct Handling
`Alley`	Most houses don't have alleys	Fill with "None"
`PoolQC`	No pool in most houses	Fill "None"
`Fence`	No fence present	Fill "None"
`MiscFeature`	Rare features (shed, elevator)	Fill "None"
`FireplaceQu`	No fireplace	Fill "None"


Before Cleaning: LotFrontage (Missing Values Present)
Count
LotFrontage
0
10
20
30
40
50
60
70
80
20
40
60
80
100
120
140


After Cleaning: LotFrontage (Median Imputed)
Count
LotFrontage
0
50
100
150
200
20
40
60
80
100
120
140

Steps of EDA- continued

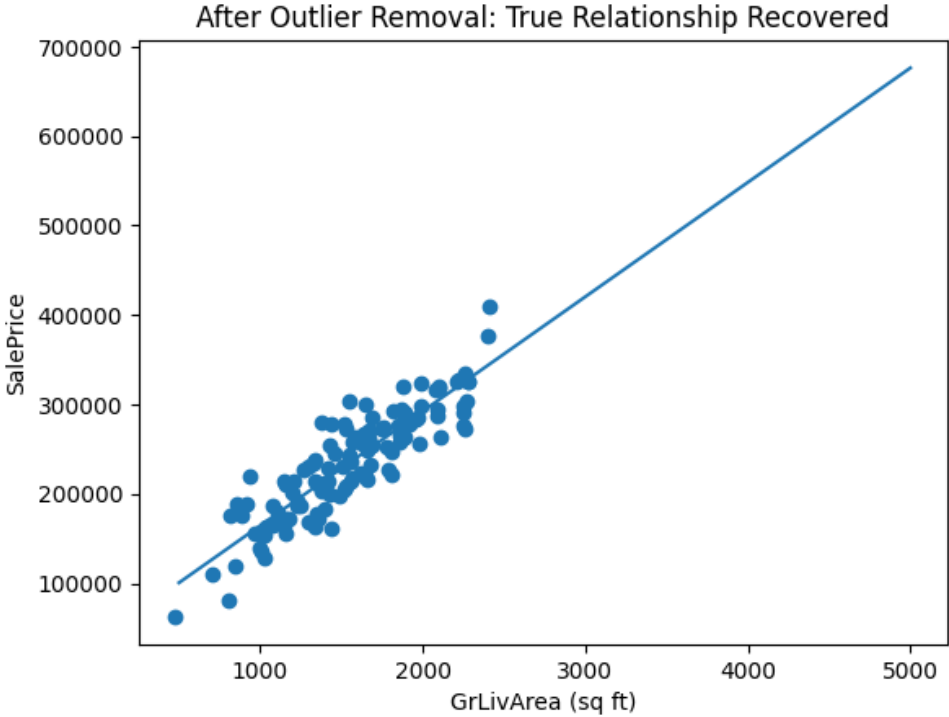
Separating Feature Types

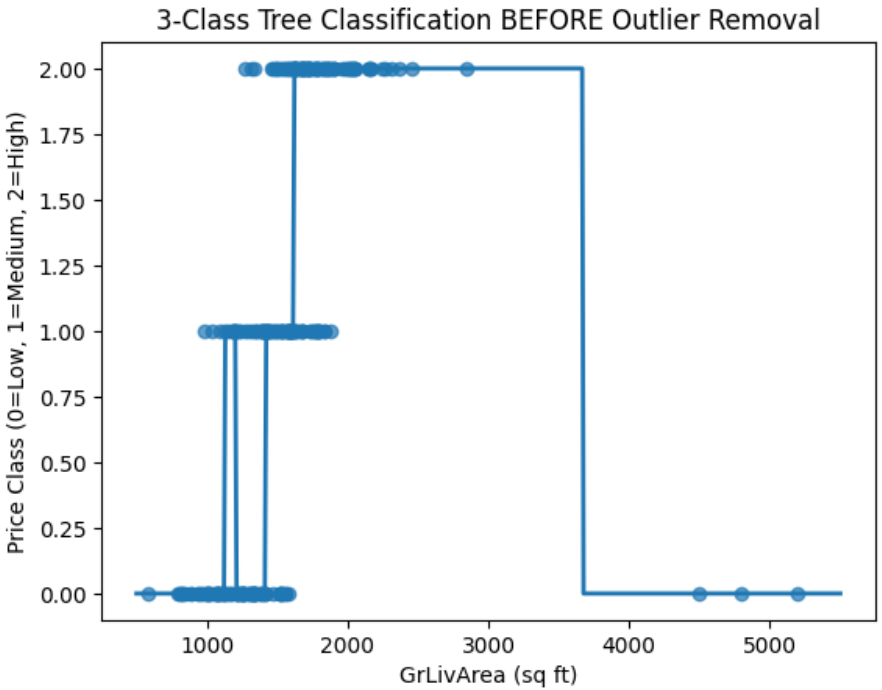
- Numerical features: continuous / discrete values
- Categorical features: nominal and ordinal
- Different preprocessing pipelines required
- Avoids incorrect transformations

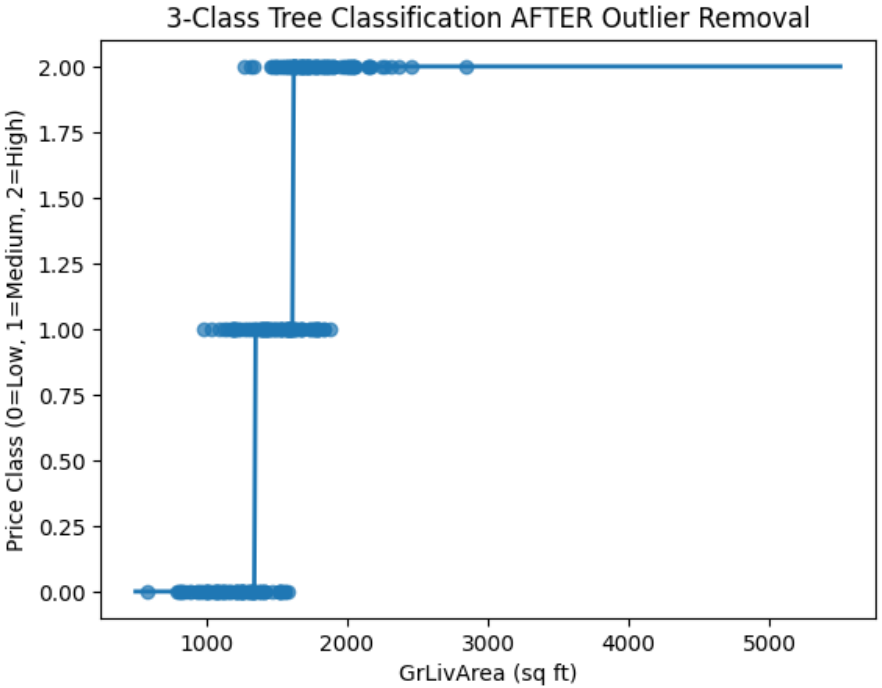
Removing or keeping outliers

- Identify extreme values
- Decide to keep, cap, or remove
- Visualize at every step.


Before Outlier Removal: Regression Line Distorted
SalePrice
GrLivArea (sq ft)
50000
100000
150000
200000
250000
300000
350000
400000
1000
2000
3000
4000
5000


After Outlier Removal: True Relationship Recovered
SalePrice
GrLivArea (sq ft)
100000
200000
300000
400000
500000
600000
700000
1000
2000
3000
4000
5000





0 = Low price | 1 = Medium price | 2 = High price

A few **extreme houses** with very large GrLivArea, Tree creates **wide, unnatural decision regions.**
Class 2 dominates a large interval Medium-price region becomes unstable and fragmented.
The tree learns rules for rare luxury houses instead of learning the typical low–medium–high

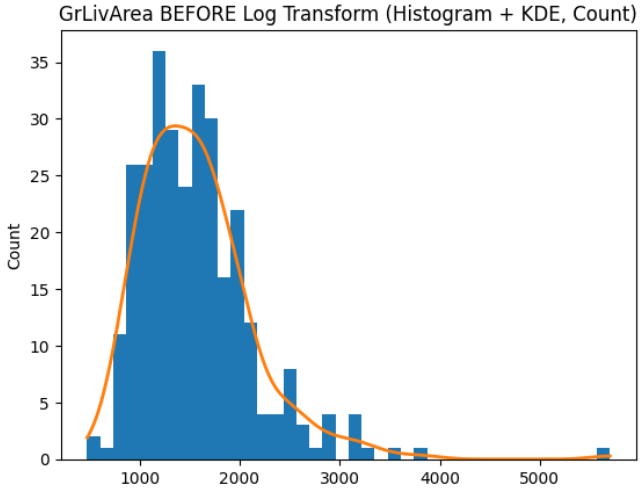
Steps of EDA- continued

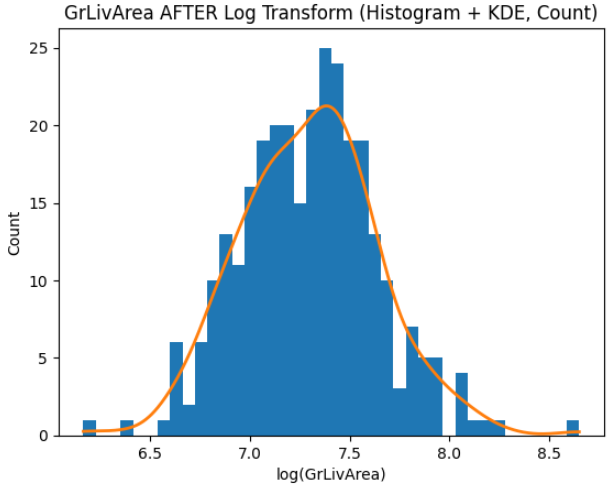
Fixing skewness (positive or negative skew)

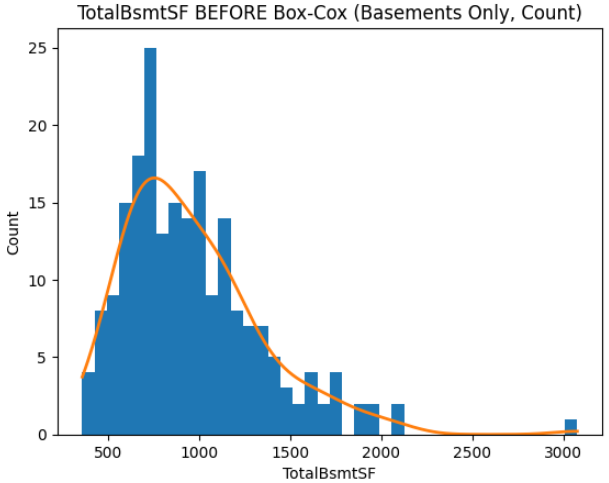
- Stabilizes variance
- Improves linearity
- Reduces influence of extreme values
- Different methods: Log, Square, Yeo –Johnson, Box-Cox

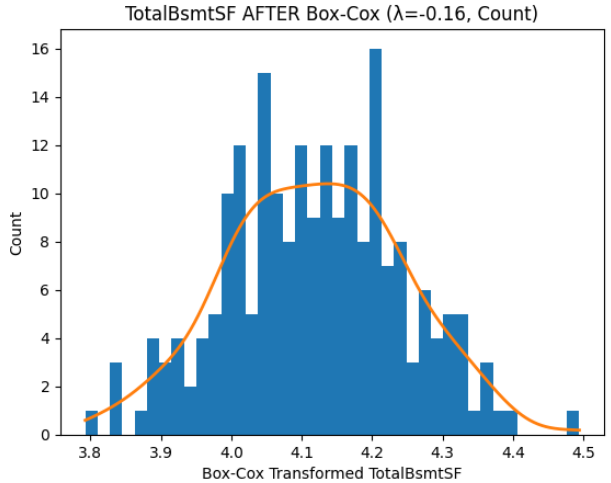
Scaling

- Scaling doesn’t change the data — it changes how the model sees the data.
- Scaling reshapes the optimization landscape while unscaled features dominate learning
- Scaling improves Interpretability of coefficients
- Improves Distance-based models (KNN, K-Means, SVM, hierarchical clustering) depend on scale
- Gradient-based models converge faster with scaled features
- Scaling doesn’t improve feature relevance, model accuracy and data quality


GrLivArea BEFORE Log Transform (Histogram + KDE, Count)
Count
0
5
10
15
20
25
30
35
1000
2000
3000
4000
5000


GrLivArea AFTER Log Transform (Histogram + KDE, Count)
Count
0
5
10
15
20
25
6.5
7.0
7.5
8.0
8.5
log(GrLivArea)


TotalBsmtSF BEFORE Box-Cox (Basements Only, Count)
Count
0
5
10
15
20
25
500
1000
1500
2000
2500
3000
TotalBsmtSF


TotalBsmtSF AFTER Box-Cox (λ=-0.16, Count)
Count
0
2
4
6
8
10
12
14
16
3.8
3.9
4.0
4.1
4.2
4.3
4.4
4.5
Box-Cox Transformed TotalBsmtSF

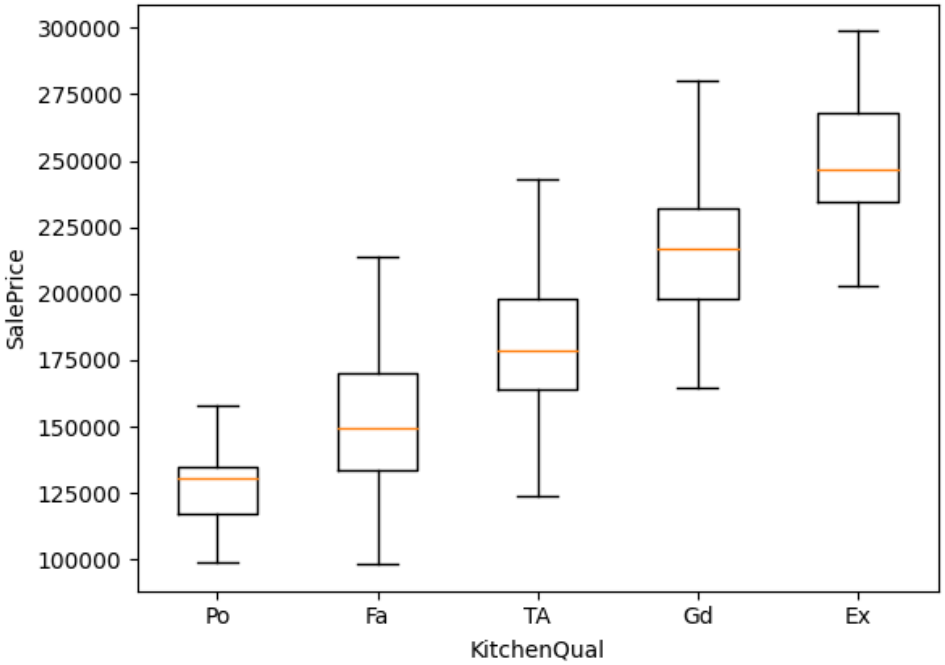
Steps of EDA- continued

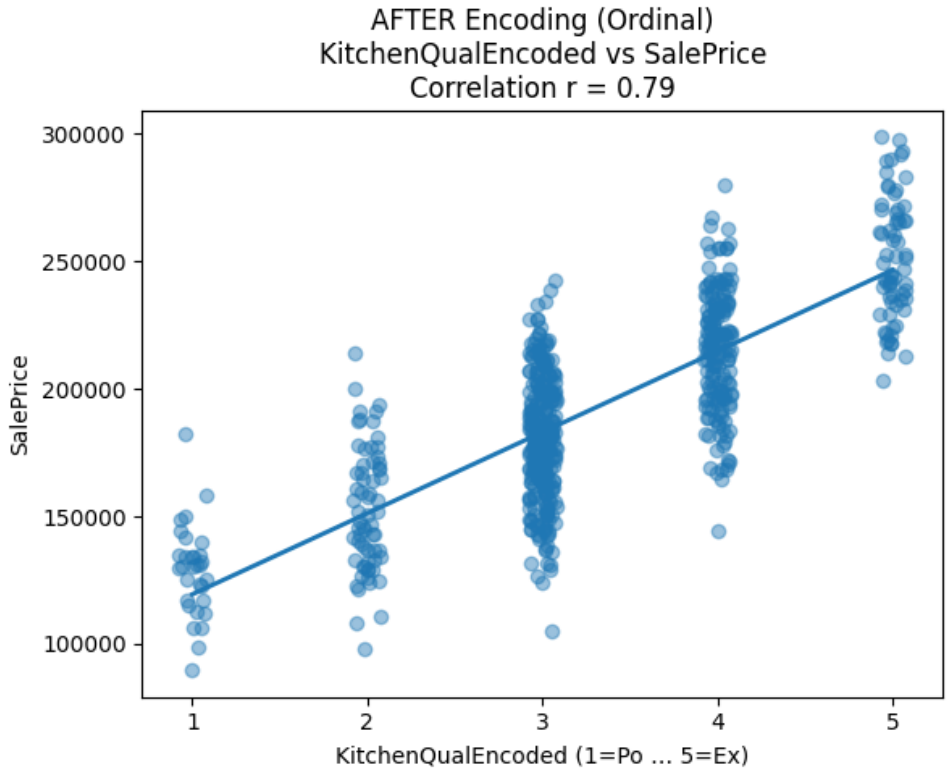
Categorical encoding

- **Without encoding,** Categorical variables remain as text, and it cannot compute:
  - Correlation with target
  - Group-wise statistics numerically
  - Multivariate relationships
- Encoding allows us to *measure* the impact of categories, not just list them.
- Most frequent imputation (mode)
- Need to handle unseen categories effectively

Feature	Encoding Choice	Why
`ExterQual`	Ordinal	Natural order
`Neighborhood`	One-Hot	No inherent order
`MSZoning`	One-Hot	Nominal

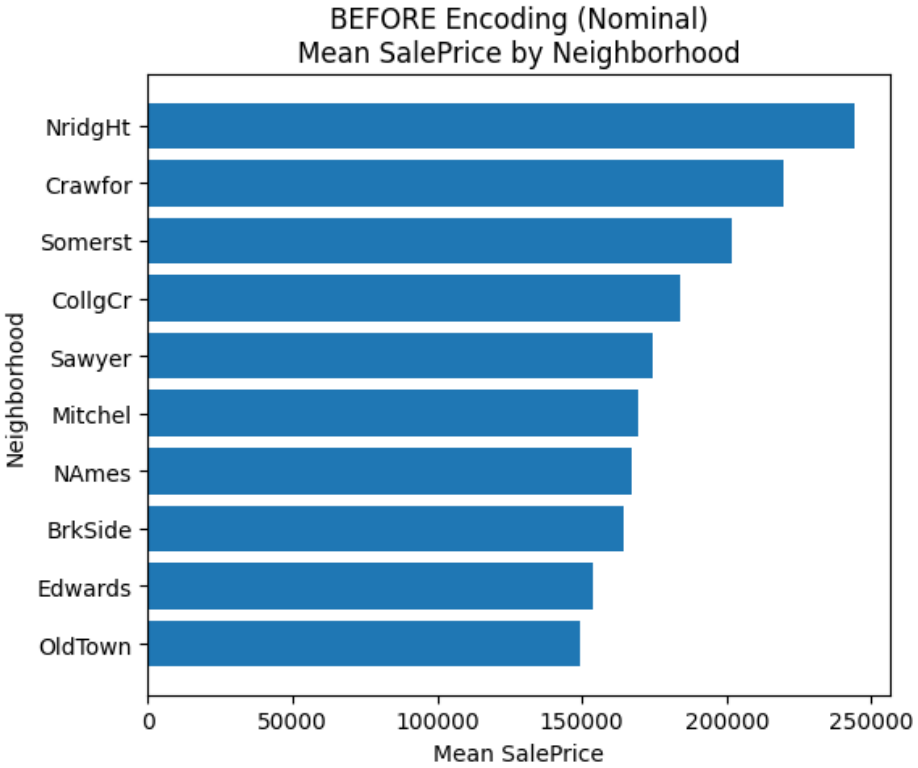
Encoding turns a visual pattern into a measurable relationship

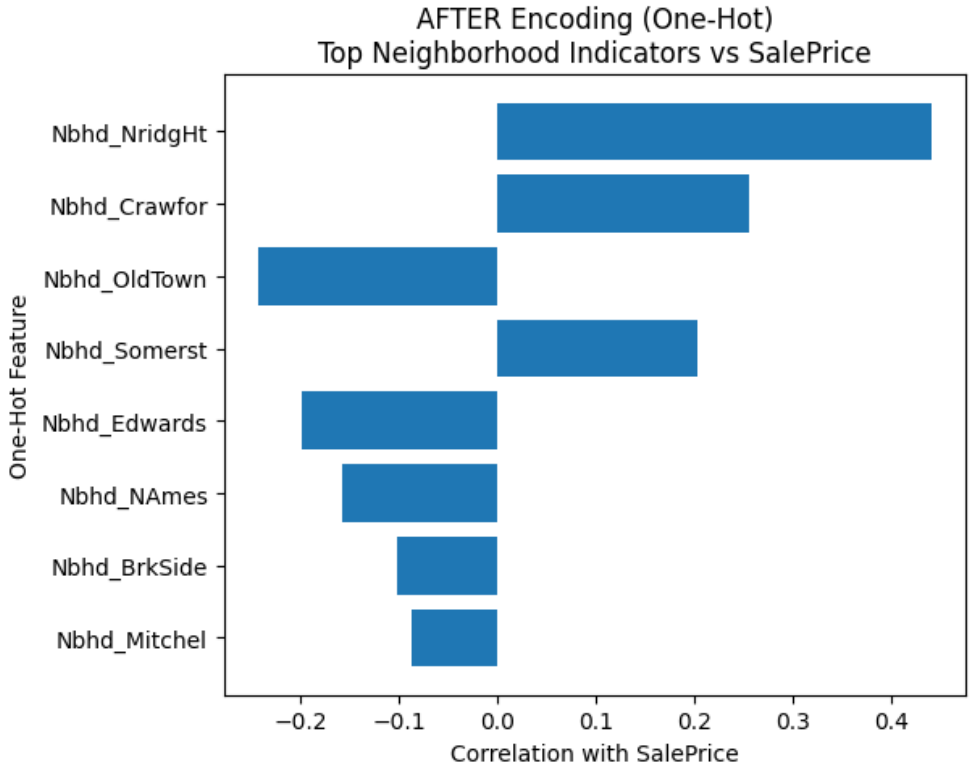




Clear ordering: Po → Fa → TA → Gd → Ex

Median price increases monotonically





Clear price stratification by area.

Some neighborhoods are consistently expensive (NridgHt, Crawfor)

Others are consistently cheaper (OldTown, Edwards)

Steps of EDA- continued

Univariate vs Bivariate vs Multivariate Statistics

EDA progresses from understanding individual variables → relationships → joint behavior.

Type	Numerical	Categorical
Central tendency	Mean, median, mode	Mode
Spread	Variance, std, IQR	Frequency
Shape	Skewness, kurtosis	Category balance
Extremes	Min, max	Rare categories

UNIVARIATE

- Typical Visualizations
  - Histogram
  - KDE
  - Boxplot
  - Bar chart (categorical)

Steps of EDA- continued

Bivariate.

Aspect	Description
What it analyzes	Relationship between **two variables**
Key questions answered	Are two variables related? How strong is the relationship? Is it linear or non-linear? Does one variable influence the other?
Common variable pairings	**Numeric ↔ Numeric** (e.g., *GrLivArea vs SalePrice*) **Categorical ↔ Numeric** (e.g., *KitchenQual vs SalePrice*) **Categorical ↔ Categorical** (e.g., *Neighborhood vs SaleCondition*)
Typical statistical measures	Correlation (Pearson, Spearman),Covariance
Typical visualizations	Scatter plot, Boxplot (category vs numeric), Line plot
Role in EDA	Identifies important relationships, guides feature selection, and motivates transformations or encoding

Multivariate

Aspect	Description
What it analyzes	Three or more variables together
Key questions answered	How variables interact jointly? Which variables matter most together? Can dimensionality be reduced?
Typical goals & methods	Checking Dependency: Multiple regression, Checking Redundancy: Correlation matrix, Clustering Prediction: Multivariate models
Typical statistical techniques	Multiple linear / logistic regression Clustering algorithms
Typical visualizations	Correlation heatmap Pair plot scatter plot
Example (House Prices)	Regression using *GrLivArea + OverallQual + Neighborhood*
Role in EDA	Explains joint effects, removes redundancy, and prepares data for modeling and dimensionality reduction

Aspect	Univariate Analysis	Bivariate Analysis	Multivariate Analysis
Key questions answered	How does the variable look like? Is it skewed? Are there outliers?	Are two variables related? How strong is the relationship? Is it linear or non-linear? Does one influence the other?	How do variables interact jointly? Which variables matter most together? Are features redundant? Can dimensionality be reduced?
Typical variable pairings	Numeric *or* categorical	Numeric ↔ NumericCategorical ↔ NumericCategorical ↔ Categorical	Numeric + categorical (mixed)Multiple numeric features
Common statistics	Mean, median, mode, Variance, std, IQR, Skewness	Correlation (Pearson, Spearman), Covariance	Multiple regression, Clustering
Typical methods	Summary statistics	Correlation analysis, Hypothesis testing	Dimensionality reduction, Dependency modelling
Example (House Prices)	Distribution of SalePrice Skewness of GrLivArea	GrLivArea vs SalePrice KitchenQual vs SalePrice	Regression using GrLivArea + OverallQual + Neighborhood on numerical features
Role in EDA	Data quality & understanding	Feature relevance & relationships	Model readiness & structure
When used in workflow	First	After univariate	After bivariate
Main risk if skipped	Garbage-in analysis	Missing key predictors	Overfitting / multicollinearity

Steps of EDA- continued

Feature engineering- how it enriches the quality of data

- **HouseAge= YrSold-YrBuilt**
  - How old the house is at the time of sale, a relative measure, not absolute
  - For EDA it is stronger, more interpretable relationship, older houses generally sell for less price
- **YearSinceRemodel=YrSold-YrRemodAdd**
  - Shows the recency of renovation and condition
  - For EDA it explains why some old houses sells high as it reveals recency bias

- **EffectiveAge = min(HouseAge, YearsSinceRemodel)**
  - It captures buyers perceived age
  - Its powerful EDA as it reduces noise and compresses multiples timelines to one signal
- Domain driven signals are important

Why use pipelines?

Pipeline in EDA is an ordered sequence of data preparation steps that are applied consistently to transform raw data into analyzable data without introducing errors or leakage.
EDA explores data; pipelines protect correctness.

Pipeline Type	Typical Steps	EDA Purpose
Numerical	Impute → Transform → Scale	Fix shape & magnitude
Categorical	Impute → Encode	Quantify categories
Mixed	Column-wise pipelines	Handle heterogeneous data
Clustering	Scale → Cluster	Segmentation

Numerical EDA pipeline: Median Imputation → Log Transform → StandardScaler
Nominal categorical pipeline: Most-Frequent Imputation → One-Hot Encoding

Key take aways

- EDA is not optional- it is foundational
  - EDA establishes data quality and validity, distributional assumptions, appropriate transformations and feasible modeling choices
  - *You cannot model what you do not understand*.
- Data representation matters more than algorithms
  - *Models don’t fail — représentations do.*
- Feature engineering boosts intepretebility
  - Reduces noise and Variance and reveal trends
  - *Good features explain the data before they predict it.*
- Simple models can perform well with good data
- EDA determines whether learning is possible; modeling determines how efficiently it happens