Assumptions

09 September 2023 08:08

- Linearity of independent variables or linear relationship between X and Y
- No endogeneity
    - Error and some independent variables are related(as the factor we omitted is included in the error term)
    - Omitted variable bias - we forgot to include a variable which has some correlation with one of the independent variables and can help predict our dependent variable better
- Normality and heteroskedasticity of errors
- No autocorrelation of errors
- No multicollinearity

Statsmodels interpretation

09 September 2023 08:30

- Durbin watson(for assumption of homoskedasticity) should be between 0 and 4
    - 2 -> no autocorrelation
    - <1 and >3 is bad
- Omnibus should be close to zero for indicating normalcy of residuals. Prob(Omnibus) should be close to 1 to indicate the same as well.
- High condition number might indicate multicollinearity.

Normalization Methods

09 September 2023 12:46

- **Min max normalization**
  - For range in [-1, 1] or [0,1]
    - $$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$$
  - For arbitrary range of [a, b]
    - $$x' = a + \frac{(x - \min(x))(b - a)}{\max(x) - \min(x)}$$
- **Mean normalization**
  - $$x' = \frac{x - \bar{x}}{\max(x) - \min(x)}$$
- **Z score normalization (Standardization)**
  - $$x' = \frac{x - \bar{x}}{\sigma}$$
- **Scaling to unit length**
  - $$x' = \frac{x}{\|x\|}$$

Error functions

09 September 2023 12:55

- L1 Regularization
  - Also known as LAD(Least absolute deviations)
    - $$S = \sum_{i=1}^{n} |y_i - f(x_i)|.$$
- L2 Regularization
  - Also known as LSE(Least Squares Error)
  - Resistant to outliers
    - $$S = \sum_{i=1}^{n} (y_i - f(x_i))^2$$
- Differences

L2 loss function	L1 loss function
Not very robust	Robust
Stable solution	Unstable solution
Always one solution	Possibly multiple solutions

L2 regularization	L1 regularization
Computational efficient due to having analytical solutions	Computational inefficient on non-sparse cases
Non-sparse outputs	Sparse outputs
No feature selection	Built-in feature selection

- R squared
  - Total sum of squares -> 1/n * summation of (y - y_pred) ^ 2
  - Total sum of square of residuals -> 1/n * summation of (y - y_mean) ^ 2
  - R squared -> 1 - (Total sum of squares / Total sum of square of residuals)
- Other metrics
  - MAE
  - MSE
  - RMSE

Feature selection methods

09 September 2023 13:14

- Feature_selection.f_regression
    - Creates simple linear regressions of each feature and the dependent variable
    - Does not take into account the correlation b/w the features

Miscellaneous

13 September 2023 08:00

- MSE = Bias ^ 2 + Variance

Fit checks

14 September 2023 08:32

- **Yellowbricks**
  - Residuals plot
  - Prediction Error plot (PredictionError from yellowbrick.regressor)
- **Seaborn**
  - sns.lmplot
- **Lime**
  - lime.lime_tabular.LimeTabularExplainer
- **SHAP**
  - shap.KernelExplainer
  - shap.summary_plot
  - Dependence plot
    - shap.dependence_plot
  - Individual force plot
    - shap.force_plot
  - Collective force plot
    - shap.force_plot
  - SHAP heatmap
    - shap.plots.heatmap
  - SHAP waterfall
    - shap.plots.waterfall
  - Feature importance
    - shap.plots.bar

Transformations

16 September 2023 08:38

- Log transformation
- Reciprocal transformation
- Square root transformation
- Exponential transformation
- Box-Cox transformation
- Yeo-Johnson transformation

Best practices

17 September 2023 18:35

Assumptions:

- The Data is model ready
- Missing Values treated accordingly
- Outlier treatment completed

Things to validate before starting a Model:

Check the data:

- Dependent Variable:
  - Distribution – Consider log to counter skewness
- Continuous Variables:
  - How are they distributed
  - Any transformations needed to convert them to a normal distribution
- Categorical:
  - How many levels are present in the data
  - Too many levels- may not help in prediction
  - Is there a need to group a few levels
- Exploratory
  - Scatter plot between DV and IDV. Look for a relationship. If they are related, is it linear or non-linear?
  - Check correlation between DV and IDVs. Check correlation between IDVs to avoid Collinearity issues
  - Check if there is sufficient variation in IDV and DV. Model cannot help much if there is less variance
- Date:
  - Is there a significance of the date field like Trend/Seasonality?
  - If so, how to include them in the model- creating variables based on the need.

Feature Engineering and Model Building:

  - Divide the data to Training – Test – Validation. Consider cross validation
  - Sample Creation: While preparing the test and train samples from the data, make sure there is no target leakage
  - Feature Engineering:
    - Avoid Target leakage. None of the engineered features should have any info (either directly or indirectly) related to your dependent variable.
    - Also, while building the features for any model, make sure you use only the variables which would be available at the time of prediction when you apply that model to data you collect in the future.

-

Regression:

Linear Regression

  - Start with an OLS model – Don't build models w/o intercept. Look for reasonable R-sq
  - Understand hypothesis of linear regression (it is a parametric test)
  - Observe the results, look for
    - Significance (Look at the P-values of the model coefficients to check the significance of the predictors. In normal practice, the p-values should be less than <0.01.
    - Relationship with dependent variable- is this making business sense

        - Collinearity (VIF) – If VIFs > 5/10. Look out and remove collinear vars.
    - Check the overall performance of the model using various factors like:
        - R-Square – should not be –ve
        - Adjusted R – square – should not be drastically different for R –sq
        - MAPE – Wherever the Mape is high see if a particular variable misspecification is causing this high MAPE
        - Plot Actual vs Predicted and look for the places where there is poor fit and identify if there was any event year on year happening or common activity happening across segments or any other information where the fit is low could be addressed
        -
    - If the number of variables is too large, reduce the variables by using the appropriate technique
        - Stepwise technique
        - PCA
        - Lasso
    - Check for the sign of coefficients and see if it makes business sense
    - Try manually adding/removing variables from the model and see how it impacts the sign and significance of other coefficients
    - Look at bivariate plots by important sub-segments and come up with interaction terms to be included in the model
    - Check the Residual Plots to check if there is any heteroskedasticity
    - If the residuals are not on the expected lines, check out the areas where the model is not predicting properly
        - See if there is a need to fit different models for different range of dependent variables
        - Check if any interactive effect can improve the overall model performance
        - Try to understand the underlying factors for poor fit in some regions. Probably need to discuss with the client and get their perspective. Do appropriate changes to improve the fit.
    - Check R^2/AdjR^2 and MAPE. If MAPE is too high, check whether errors on smaller values are inflating MAPE. Consider taking a Median
    - Check model performance in key sub-segments

Other Regression Algorithms (RF, GBM, XGBoost)

- There is no assumption like Linear regression
- Build a model and compare it with baseline linear regression
- Use train/validation/test split or cross validation / test split
- Algorithms like RF, GBM, XGBoost are prone to overfitting. Avoid this by comparing the performance across train, validation and test sets
- See variable importance table to understand key drivers impacting the outcome
- Use Shap/Lime based model interpretation methods to understand key drivers and the directional impact of each feature on dependent variable. Ensure that this relationship is making business sense, validate these results with your lead or client.
- Since by default trees take encoded categorical columns as numerical columns, either use one hot encoding or try using catboost if there are many categorical independent variables