MODEL REPORT

Model: <<modelop.storedModel.modelMetaData.name>>

### Description: <<modelop.storedModel.modelMetaData.description>>

# Model Information

|  |  |
| --- | --- |
| Item | Value |
| Model ID | <<modelop.storedModel.id>> |
| Model Name | <<modelop.storedModel.modelMetaData.name>> |
| Business Unit | <<modelop.storedModel.modelMetaData.modelOrganization>> |
| Model Category | <<modelop.storedModel.modelMetaData.modelUseCategory>> |
|  |  |
| Risk Tier | <<modelop.storedModel.modelMetaData.modelRisk>> |
| Model Creation Date | <<modelop.storedModel.createdBy>> |

# Key Personnel

|  |  |
| --- | --- |
| Item | Value |
| Business Owner | Jane Smith |
| Model Developer | <<modelop.storedModel.createdBy>> |
| Governance Executive | Jim Smith |
| Model Risk Officer | James Dean |
| Model Validator | John Doe |
| IT Owner | Jon Goode |

**Executive Summary**

*Provide a concise summary description of the model and its proposed use(s), model results, and key conclusions from the model review process.*

**Statistical Analysis / Testing Summary**

|  |  |  |
| --- | --- | --- |
| Category | Passes | Reason |
| Performance | <<modelop.modelTestResult.dmnRuleResults.[testCategory=Performance].passes>> | <<modelop.modelTestResult.dmnRuleResults.[testCategory=Performance].reason>> |
| Data Drift - Kolmogorov Smirnov | <<modelop.modelTestResult.dmnRuleResults.[testCategory=KS Data Drift].passes>> | <<modelop.modelTestResult.dmnRuleResults.[testCategory=KS Data Drift].reason>> |
| Concept Drift - Kolmogorov Smirnov | <<modelop.modelTestResult.dmnRuleResults.[testCategory=KS Concept Drift].passes>> | <<modelop.modelTestResult.dmnRuleResults.[testCategory=KS Concept Drift].reason>> |
| Characteristic Stability | <<modelop.modelTestResult.dmnRuleResults.[testCategory=Characteristic Stability].passes>> | <<modelop.modelTestResult.dmnRuleResults.[testCategory=Characteristic Stability].reason>> |
| Bias Disparity | <<modelop.modelTestResult.dmnRuleResults.[testCategory=Bias Disparity].passes>> | <<modelop.modelTestResult.dmnRuleResults.[testCategory=Bias Disparity].reason>> |
| Autocorrelation | <<modelop.modelTestResult.dmnRuleResults.[testCategory=Autocorrelation].passes>> | <<modelop.modelTestResult.dmnRuleResults.[testCategory=Autocorrelation].reason>> |
| Homoscedacticity: Breusch Pagan | <<modelop.modelTestResult.dmnRuleResults.[testCategory=Homoscedacticity: Breusch Pagan].passes>> | <<modelop.modelTestResult.dmnRuleResults.[testCategory=Homoscedacticity: Breusch Pagan].reason>> |
| Homoscedacticity: Engle Lagrange Multiplier | <<modelop.modelTestResult.dmnRuleResults.[testCategory=Homoscedacticity: Engle Lagrange Multiplier].passes>> | <<modelop.modelTestResult.dmnRuleResults.[testCategory=Homoscedacticity: Engle Lagrange Multiplier].reason>> |
| Homoscedacticity: Ljung Box Q | <<modelop.modelTestResult.dmnRuleResults.[testCategory=Homoscedacticity: Ljung Box Q].passes>> | <<modelop.modelTestResult.dmnRuleResults.[testCategory=Homoscedacticity: Ljung Box Q].reason>> |
| Normality: Kolmogorov Smirnov | <<modelop.modelTestResult.dmnRuleResults.[testCategory=Normality: Kolmogorov Smirnov].passes>> | <<modelop.modelTestResult.dmnRuleResults.[testCategory=Normality: Kolmogorov Smirnov].reason>> |
| Normality: Anderson Darling | <<modelop.modelTestResult.dmnRuleResults.[testCategory=Normality: Anderson Darling].passes>> | <<modelop.modelTestResult.dmnRuleResults.[testCategory=Normality: Anderson Darling].reason>> |
| Normality: Cramer Von Mises | <<modelop.modelTestResult.dmnRuleResults.[testCategory=Normality: Cramer Von Mises].passes>> | <<modelop.modelTestResult.dmnRuleResults.[testCategory=Normality: Cramer Von Mises].reason>> |
| Linearity: Pearson Correlation | <<modelop.modelTestResult.dmnRuleResults.[testCategory=Linearity: Pearson Correlation].passes>> | <<modelop.modelTestResult.dmnRuleResults.[testCategory=Linearity: Pearson Correlation].reason>> |
| Mulitcolinearity | <<modelop.modelTestResult.dmnRuleResults.[testCategory=Mulitcolinearity].passes>> | <<modelop.modelTestResult.dmnRuleResults.[testCategory=Mulitcolinearity].reason>> |

**Statistical Analysis**

**Performance Metrics**

*Provide a textual analysis of the metrics and what they indicate*

|  |
| --- |
| <<modelop.modelTestResult.testResults.(performance)[0].values>> |

Table 1 - Metrics

**Stability Analysis**

*Provide a textual analysis of the stability analysis and what they indicate*

<<modelopgraph.stability.\*>>

**Statistical Diagnostics**

*Below is a detailed list of the standardized statistical diagnostics tests for the model.*

Autocorrelation Metrics:

|  |  |
| --- | --- |
| *Durbin-Watson Test:* |  |
| dw\_statistic | <<modelop.modelTestResult.testResults.(dw\_statistic)>> |

Normality Metrics:

|  |  |  |
| --- | --- | --- |
| Test | Metric | Value |
| Kolmogorov-Smirnov | ks\_statistic | <<modelop.modelTestResult.testResults.(ks\_statistic)>> |
| Kolmogorov-Smirnov | ks\_p\_value | <<modelop.modelTestResult.testResults.(ks\_p\_value)>> |
| Cramer-von Mises | cvm\_statistic | <<modelop.modelTestResult.testResults.(cvm\_statistic)>> |
| Cramer-von Mises | cvm\_p\_value | <<modelop.modelTestResult.testResults.(cvm\_p\_value)>> |
| Anderson-Darling | ad\_statistic | <<modelop.modelTestResult.testResults.(ad\_statistic)>> |
| Anderson-Darling | ad\_p\_value | <<modelop.modelTestResult.testResults.(ad\_p\_value)>> |

Linearity Metrics:

*Pearson Correlation*

|  |
| --- |
| <<modelop.modelTestResult.testResults.(linearity)[0].values>> |

Homoscedasticity Metrics:

|  |  |  |
| --- | --- | --- |
| Test | Metric | Value |
| Ljung-Box Q Test | minLjungBoxQPValue | <<modelop.modelTestResult.testResults.(Homoscedasticity\_minLjungBoxQPValue)>> |
| Ljung-Box Q Test | maxLjungBoxQPValue | <<modelop.modelTestResult.testResults.(Homoscedasticity\_maxLjungBoxQPValue)>> |
| Engle's Lagrange Multiplier | engle\_lm\_statistic | <<modelop.modelTestResult.testResults.(engle\_lm\_statistic)>> |
| Engle's Lagrange Multiplier | engle\_lm\_statistic | <<modelop.modelTestResult.testResults.(engle\_lm\_p\_value)>> |
| Breusch-Pagan | breusch\_pagan\_lm\_statistic | <<modelop.modelTestResult.testResults.(breusch\_pagan\_lm\_statistic)>> |
| Breusch-Pagan | breusch\_pagan\_lm\_p\_value | <<modelop.modelTestResult.testResults.(breusch\_pagan\_lm\_p\_value)>> |
| Breusch-Pagan | breusch\_pagan\_f\_statistic | <<modelop.modelTestResult.testResults.(breusch\_pagan\_f\_statistic)>> |
| Breusch-Pagan | breusch\_pagan\_f\_p\_value | <<modelop.modelTestResult.testResults.(breusch\_pagan\_f\_p\_value)>> |

Variance Inflation Factors:

*Variance Inflation Factors:*

|  |
| --- |
| <<modelop.modelTestResult.testResults.(multicollinearity)[0].values>> |

**Ethical Fairness Analysis**

*Provide a textual analysis of bias results and control measures put in place to eliminate the bias in the provided data and results.*

**Bias Analysis for Protected Classes**

|  |  |
| --- | --- |
| Metric | Value |
| Bias\_minPPRDisparityValue | <<modelop.modelTestResult.testResults.(Bias\_minPPRDisparityValue)>> |
| Bias\_minPPRDisparityValueFeature | <<modelop.modelTestResult.testResults.(Bias\_minPPRDisparityValueFeature)>> |
| Bias\_maxPPRDisparityValue | <<modelop.modelTestResult.testResults.(Bias\_maxPPRDisparityValue)>> |
| Bias\_maxPPRDisparityValueFeature | <<modelop.modelTestResult.testResults.(Bias\_maxPPRDisparityValueFeature)>> |