

1. **Problem Statement:** Predicting insurance charges based on input variables such as age, sex, BMI, number of children, and smoking status, with insurance charges as the output variable.

2. **Basic Info about Dataset:**

- Total number of rows and columns not mentioned.

Input Variables:

- Age (continuous)
- Sex (categorical)
- BMI (continuous)
- Children (continuous)
- Smoker (categorical)

Output Variable:

- Charges (continuous)

3. **Pre-processing Method:** One-hot encoding using the `get_dummies` method in Python for converting categorical variables into numerical form.

4. **Model Development with R-squared- SVM model:**

- Support Vector Machine (SVM)
 - Various hyperparameters (C values) tested for different kernels: Linear, RBF, Polynomial, and Sigmoid.
 - R-squared values documented for each model.

S.No	Hyper Parameter	Linear	RBF (Non-Linear)	Polynomial	Sigmoid
1	C10	0.46593	0.0331	0.03906	-7.33
2	C100	0.63124	0.3137	0.6156	-0.01756
3	C500	0.7644	0.6619	0.8234	0.1254
4	C1000	0.7671	0.8114	0.852	0.287
5	C2000	0.7478	0.8583	0.85623	-0.5878
6	C5000	0.74536	0.8784	0.8538	-7.33

5. **Research Values (R-squared scores):**

6. The RBF kernel consistently improves R-squared value with increasing C values, peaking at C5000.

7. **Final Model:** The chosen final model utilizes the RBF kernel with C=5000, achieving the highest R-squared value of 0.8784. This model was selected for its superior performance compared to other hyperparameters and kernels, ensuring a better fit to the data and potentially higher prediction accuracy.

Model Development with R-squared - Decision Tree Model:

Decision Tree (DT)

R- Research Values (R-squared scores):

- The CRITERION, MAX FEATURES, and SPLITTER consistently improve R-squared value with increasing R Squared values.

The model configuration with max features set to 'auto' and splitter set to 'random' achieved an R value of 0.74420, demonstrating a notable level of performance.

Final Model:

The chosen model configuration with the criterion set to 'Mae', max features set to 'auto', and splitter set to 'random' achieved an R value of 0.74420, demonstrating a strong level of performance. This configuration was selected for its notable performance, ensuring a reliable fit to the data and potentially higher prediction accuracy.

S.No	CRITERION	MAX FEATURES	SPLITTER	R VALUE
1	Mse	auto	best	0.7083
2	Mse	auto	random	0.6782
3	Mse	sqrt	best	0.6488
4	Mse	sqrt	random	0.6307
5	Mse	log2	best	0.7075
6	Mse	log2	random	0.5103
7	Mae	auto	best	0.6787
8	Mae	auto	random	0.74420
9	Mae	sqrt	best	0.7050
10	Mae	sqrt	random	0.6272
11	Mae	log2	best	0.6398
12	Mae	log2	random	0.6973
13	frideman_mse	auto	best	0.68097
14	frideman_mse	auto	random	0.7139
15	frideman_mse	sqrt	best	0.70823
16	frideman_mse	sqrt	random	0.7041
17	frideman_mse	log2	best	0.7486
18	frideman_mse	log2	random	0.6848

Interpreting the Hyperparameter Tuning Results for Random Forest:

The provided table summarizes the performance of various Random Forest models with different hyperparameters. The key metric for performance is the R-squared value, with higher values indicating better model performance.

Best Hyperparameters:

From the table, we identify the configurations with the highest R-squared values:

A	B	C	D	E	F	G	H	I
S.No	n_estimators	max_features	max_depth	min_samples_split	min_samples_leaf	bootstrap	criterion	R-squared
1	100	auto	None	2	1	TRUE	mse	0.82905
2	100	sqrt	None	2	1	TRUE	mse	0.8669
3	100	log2	None	2	1	TRUE	mse	0.8409
4	200	auto	20	2	1	TRUE	mse	0.8369
5	200	sqrt	20	2	1	TRUE	mse	0.8256
6	200	log2	20	2	1	TRUE	mse	0.7466
7	200	auto	None	10	4	TRUE	mse	0.8329
8	200	sqrt	None	10	4	TRUE	mse	0.8274
9	200	log2	None	10	4	TRUE	mse	0.8141
10	500	auto	20	2	1	FALSE	mae	0.8315
11	500	sqrt	20	2	1	FALSE	mae	0.8285
12	500	log2	20	2	1	FALSE	mae	0.8252
13	500	auto	None	10	4	FALSE	mae	0.8104
14	500	sqrt	None	10	4	FALSE	mae	0.8138
15	500	log2	None	10	4	FALSE	mae	0.801

• **Model 2:**

- n_estimators: 100
- max_features: sqrt
- max_depth: None
- min_samples_split: 2
- min_samples_leaf: 1
- bootstrap: TRUE
- criterion: mse
- R-squared: 0.8669

Final Model Selection:

Based on the highest R-squared value, Model 2 is selected as the final model for deployment.

Summary of the Selected Model:

- Number of Estimators: 100
- Max Features: sqrt (square root of the total number of features)
- Max Depth: None
- Min Samples Split: 2
- Min Samples Leaf: 1
- Bootstrap: TRUE
- Criterion: mse (Mean Squared Error)
- R-squared: 0.8669

This model configuration is chosen due to its superior performance in terms of the R-squared value, indicating a better fit to the data and hence, more reliable predictions.

MultiLinear Regression Model:

In this regression model, identified the following coefficients:

- The first coefficient is approximately 257.80.
- The second coefficient is approximately 321.06.
- The third coefficient is approximately 469.58.
- The fourth coefficient is approximately -41.75.
- The fifth coefficient is approximately 23418.67.

Coefficients: These numbers represent how much each independent variable affects the predicted outcome. For example, a higher coefficient means that variable has a stronger impact.

Intercept: This value (-12057.24) is what the model predicts when all independent variables are zero. It's like the starting point of our predictions.

R-squared: This is a measure of how well our model fits the data. The higher the R-squared value (0.7895 in this case), the better our model explains the variation in the predicted outcome. So, around 78.95% of the variability in the outcome can be explained by our model's variables.

Conclusion:

The Random Forest model was selected with specific settings to predict insurance charges:

1. **Number of Trees (n_estimators):** 100 trees were used to make predictions. Having more trees often leads to better predictions, though it takes more time to train the model.

2. **Maximum Features (max_features)**: The square root of the total number of features was considered for each split in the trees. This helps make the model more diverse and robust.
3. **Maximum Depth (max_depth)**: Trees were allowed to grow as deep as possible until each leaf node is pure or contains only a few samples. This flexibility helps capture complex relationships in the data.
4. **Minimum Samples to Split (min_samples_split)**: At least 2 samples were required in a node to split it further. This helps prevent overfitting by ensuring that splits are meaningful.
5. **Minimum Samples per Leaf (min_samples_leaf)**: At least 1 sample was required in each leaf node. This ensures that each prediction is based on at least some data.
6. **Bootstrap Samples (bootstrap)**: Bootstrapping was used, meaning data was sampled with replacement when building each tree. This adds randomness to the model and helps reduce overfitting.
7. **Split Criterion (criterion)**: Mean Squared Error (MSE) was used to measure the quality of each split. This criterion helps the model find the best way to split the data at each node.

The Random Forest model, selected due to its superior R-squared value compared to other models such as Decision Trees, Multilinear Regression, and Support Vector Machine, has been chosen for deployment. A pickle file has been generated for this model, and its predicted outputs will be deployed in the production environment.

With these settings, the Random Forest model achieved an R-squared value of 0.8669, meaning it explains around 86.69% of the variability in insurance charges based on the chosen features. This indicates that the model performs well in predicting insurance charges based on the given data.