Answers (id ffs analysis.docx)

Following is the result after each iteration where a model with least R2 or RSE is selected.

The model parameter is evaluated using training data.

Iteration - 1

{'features': ['cement'], 'score': 0.2419698823177734, 'RSE': 14.714857553166272}

Iteration - 2 with ‘cement’ as the selected feature

{'features': ['cement', 'superplasticizer'], 'score': 0.3455138491756006, 'RSE': 13.662982462120675}

Iteration - 3 with ‘cement’ + ‘superplasticizer’ as the selected feature set

{'features': ['cement', 'superplasticizer', 'age'], 'score': 0.4695808011072409, 'RSE': 12.28457257839711}

Iteration - 4

{'features': ['cement', 'superplasticizer', 'age', 'blast\_furnace\_slag'], 'score': 0.5424207142076762, 'RSE': 11.39790465333994}

Iteration - 5

{'features': ['cement', 'superplasticizer', 'age', 'blast\_furnace\_slag', 'water'], 'score': 0.5741064470431447, 'RSE': 10.989894412884945}

Iteration - 6

{'features': ['cement', 'superplasticizer', 'age', 'blast\_furnace\_slag', 'water', 'fly\_ash'], 'score': 0.6041998313162502, 'RSE': 10.587840884855513}

Iteration - 7

{'features': ['cement', 'superplasticizer', 'age', 'blast\_furnace\_slag', 'water', 'fly\_ash', 'coarse\_aggregate'], 'score': 0.6047054598231376, 'RSE': 10.580955089655038}

Iteration - 8

{'features': ['cement', 'superplasticizer', 'age', 'blast\_furnace\_slag', 'water', 'fly\_ash', 'coarse\_aggregate', 'fine\_aggregate'], 'score': 0.6066613439796213, 'RSE': 10.554276994619395}

Running the model against the test data results in the following score.

Iteration - 1

{'features': ['cement'], 'score': 0.2629075826701883, 'RSE': 13.682523927294815}

Iteration - 2

{'features': ['cement', 'superplasticizer'], 'score': 0.36562473119919703, 'RSE': 12.68242845987156}

Iteration - 3

{'features': ['cement', 'superplasticizer', 'age'], 'score': 0.512467099143326, 'RSE': 11.097278412430631}

Iteration - 4

{'features': ['cement', 'superplasticizer', 'age', 'blast\_furnace\_slag'], 'score': 0.5726703132604456, 'RSE': 10.377641175729796}

Iteration - 5

{'features': ['cement', 'superplasticizer', 'age', 'blast\_furnace\_slag', 'water'], 'score': 0.6089099351754887, 'RSE': 9.919311136182783}

Iteration - 6

{'features': ['cement', 'superplasticizer', 'age', 'blast\_furnace\_slag', 'water', 'fly\_ash'], 'score': 0.6374040663473707, 'RSE': 9.54349496917716}

Iteration - 7

{'features': ['cement', 'superplasticizer', 'age', 'blast\_furnace\_slag', 'water', 'fly\_ash', 'coarse\_aggregate'], 'score': 0.6362444187327454, 'RSE': 9.559078277413327}

Iteration - 8

{'features': ['cement', 'superplasticizer', 'age', 'blast\_furnace\_slag', 'water', 'fly\_ash', 'coarse\_aggregate', 'fine\_aggregate'], 'score': 0.6352987285077925, 'RSE': 9.571767652109164}

**Result:**

The best model that fits best the test data is with the following features.

{'features': ['**cement**', ‘**superplasticizer** ', '**age**', '**blast\_furnace\_slag**', '**water**', '**fly\_ash**', '**coarse\_aggregate**'], 'score': 0.6362444187327454, 'RSE': 9.559078277413327}

The model doesn’t depend on “**fine\_aggregate**” much