ECE 633

Regression Analysis of Predicting Deviation from highest grade Sweet Potato for SweetAPPS Research project

Advisor:
Dr. Cranos Williams

Presented by:
Rashmi Datta
MS, Electrical and Computer Engineering

The Dataset

Sweet Potato data

3D Reconstruction

- AxialLength
- TipLength
- Curvature
- MaxDiameter
- LWRatio
- TailLength
- BodyLength
- TailBodyRatio,
- Volume
- AverageCrossSectionRadius
- Weight

Grading/User-defined

- Roundness
- Tailedness
- Blockiness
- Curviness
- Deviation from US No1

Templating/Shape Metrics

- firstorder_count
- tail_count
- residerror_count
- roothairs_count
- sphereerrors_count area
- Length(px, in)
- Width(px, in)
- Weight

The Dataset

- Our research works with the data resulting from the high-throughput imagery approach to sweet potato shape and size detection (resulting from Haque et al. 2021,Ref [1]), combined with our subjective user scoring project, put together by Daniel Perondi.

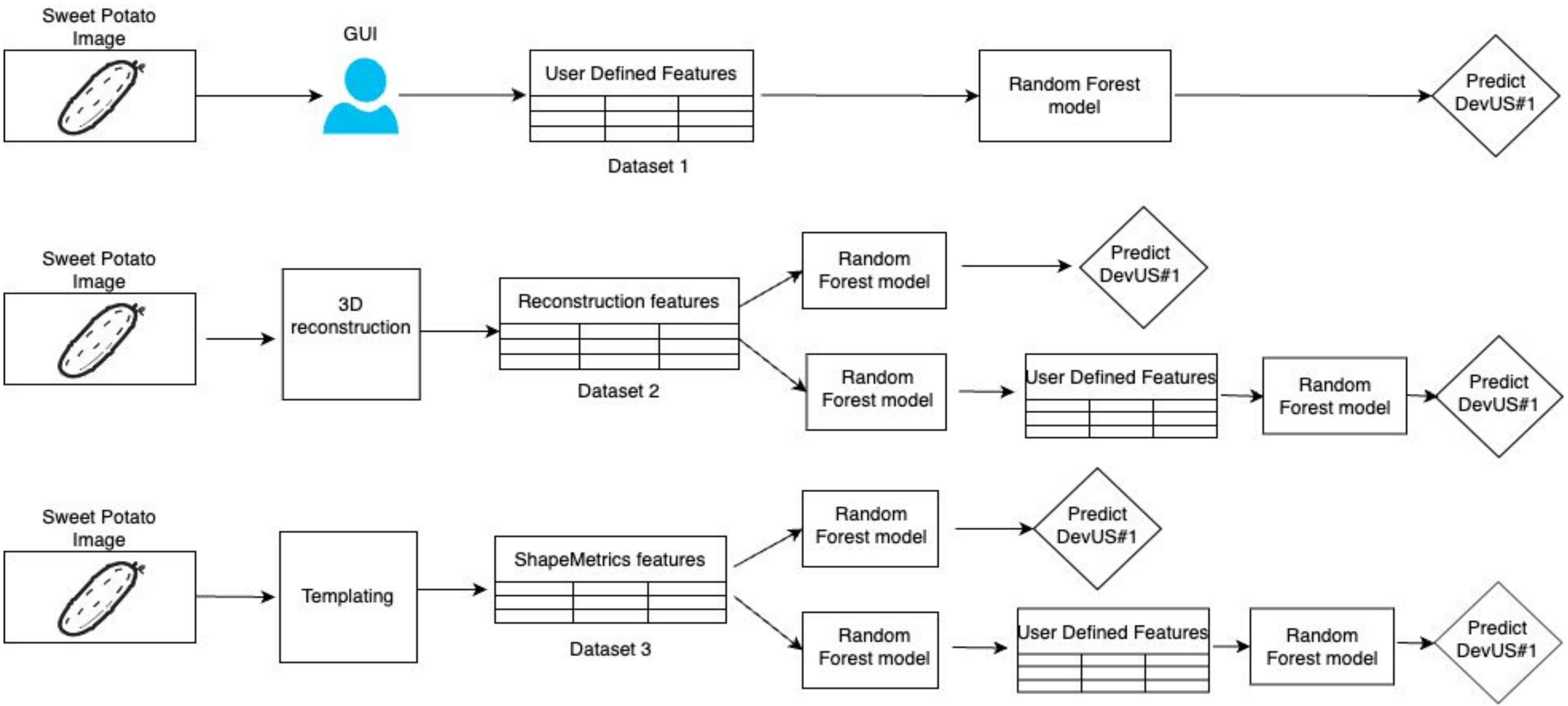
  - The *objective*, image-based data is in the dataset called "***reconstruction***".
  - The *subjective*, classification data is in the dataset called "***grading***".
  - The "***shapeMetrics"*** dataset is compiled by Michael Kudenov's results(Ref [3]) of the templating algorithm.

Reference literature & our problem statement

Algorithm	• Binary Classification	• Regresion
Dataset	• 1300 images graded good or cull	• 2000 images each graded by > 3 people • reconstruction data based on ams.usda.gov/grades-standards • templating /shape metrics data compiled by Dr.Kudenov
Models used	• Neural Network • Logistic Regression • Random Forest • Bayesian Network • Decision Tree	• Lasso • Ridge • Elastic Net • Random Forest • SVR

Ref [1]

Breakdown of our Problem Statement



Primary objectives

Predict Deviation from US#1 from **user-defined features**

Predict Deviation from US#1 from **extracted features**

Predict Deviation from US#1 from **templating features**

Describe each of the **user-defined** features in terms of combination of the **extracted features**

Describe each of the **user-defined** features in terms of combination of the **templating features**

Assess Performance between the 3 datasets - which predicts the best?

Models used for evaluation

- **Lasso Regression**
  - minimizes the sum of two components: Residual Sum of Squares (RSS) and Penalty Term (also called the "*L1 regularization term*").
- **Ridge Regression**
  - adds *L2 regularization* to *mitigate multicollinearity* (high correlation between independent variables/features) and prevents overfitting.
- **Elastic Net Regression**
  - combines the principles of both Ridge Regression and Lasso Regression, introduces both L1 (Lasso) and L2 (Ridge) regularization terms into the linear regression objective function.
- **Random Forest Regression**
  - combines *multiple decision trees* to make predictions, through an *ensemble learning technique* used for regression tasks
- **Support Vector Regression(SVR)**
  - extends the principles of Support Vector Machines to regression, aims to minimize the error between the predicted values f(x) and the actual target values y, while also considering a tolerance margin or "epsilon tube"(ε-tube) around the regression line

Evaluation

- ❖ In the following slides, we have used various metrics (such as ***MSE*** and ***R-square***) to evaluate the performances of each of the five regression models across all the three datasets-user defined, reconstruction and templating.
- ❖ Then, we compared their results to find our best performing/ champion model for further analysis.

NC STATE UNIVERSITY

Methods and analysis

	Lasso	Ridge	Elastic Net	SVR	Random Forest
devUS#1 from grading	**MSE**: training set 299.8, test set 354.0 **R squared**: training set 0.54, test set 0.5	**MSE**: training set 295.24, test set 357.26 **R squared**: training set 0.54, test set 0.5	**MSE:** training set 331.86, test set 386.01 **R squared**: training set 0.54, test set 0.5	**MSE** training set 307.35, test set 367.96 **R squared:** training set 0.52, test set 0.48	**Best Hyperparameters**: {'max_depth': 10, 'min_samples_leaf': 4, 'min_samples_split': 2, 'n_estimators': 100} **Average MSE** (Cross-Validation): 265.51
devUS#1 from reconstruction	**MSE**: training set 572.91, test set 650.31 **R squared**: training set 0.16, test set 0.13	**MSE**: training set 533.78, test set 612.88 **R squared**: training set 0.17, test set 0.14	**MSE**: training set 587.88, test set 660.75 **R squared**: training set 0.17, test set 0.14	**MSE:** training set 599.99, test set 713.16 **R squared:** training set 0.07, test set 0.0	**Best Hyperparameters**: {'max_depth': 10, 'min_samples_leaf': 4, 'min_samples_split': 10, 'n_estimators': 150} **Average MSE** (Cross-Validation): 608.99
roundness from reconstruction	**MSE**: training set 345.93, test set 399.83 **R squared**: training set 0.64, test set 0.53	**MSE**: training set 329.78, test set 405.03 **R squared**: training set 0.64, test set 0.54	**MSE**: training set 360.44, test set 406.52 **R squared**: training set 0.63, test set 0.54	**MSE:** training set 352.73, test set 419.85 **R squared:** training set 0.61, test set 0.51	**Best Hyperparameters**: {'max_depth': 10, 'min_samples_leaf': 4, 'min_samples_split': 10, 'n_estimators': 150} **Average MSE** (Cross-Validation): 351.45
tailedness from reconstruction	**MSE**: training set 780.33, test set 764.04 **R squared**: training set 0.33, test set 0.29	**MSE**: training set 722.12, test set 712.81 **R squared**: training set 0.33, test set 0.28	**MSE**: training set 823.09, test set 798.64 **R squared**: training set 0.33, test set 0.28	**MSE:** training set 772.29, test set 777.4 **R squared:** training set 0.28, test set 0.22	**Best Hyperparameters**: {'max_depth': None, 'min_samples_leaf': 4, 'min_samples_split': 2, 'n_estimators': 100} **Average MSE** (Cross-Validation): 866.73
blockiness from reconstruction	**MSE**: training set 547.94, test set 518.69 **R squared**: training set 0.19, test set 0.2	**MSE**: training set 519.16, test set 487.3 **R squared**: training set 0.19, test set 0.2	**MSE**: training set 556.86, test set 528.75 **R squared**: training set 0.2, test set 0.2	**MSE:** training set 547.37, test set 524.76 **R squared:** training set 0.15, test set 0.14	**Best Hyperparameters**: {'max_depth': 20, 'min_samples_leaf': 4, 'min_samples_split': 2, 'n_estimators': 100} **Average MSE** (Cross-Validation): 563.38
curviness from reconstruction	**MSE**: training set 449.36, test set 596.81 **R squared**: training set 0.16, test set 0.13	**MSE**: training set 425.57, test set 564.8 **R squared**: training set 0.16, test set 0.13	**MSE**: training set 455.0, test set 601.78 **R squared**: training set 0.15, test set 0.12	**MSE:** training set 507.71, test set 705.47 **R squared:** training set 0.0, test set -0.08	**Best Hyperparameters**: {'max_depth': 10, 'min_samples_leaf': 4, 'min_samples_split': 2, 'n_estimators': 50} **Average MSE** (Cross-Validation): 493.01
yvar=['devUS#1'] (Predict from Prediction)	**MSE:** training set 639.02 MSE test set 721.26 **R squared:** training set 0.07, test set 0.06	**MSE**: training set 623.52 MSE test set 704.24 **R squared:** training set 0.06, test set 0.03	**MSE:** training set 643.13 MSE test set 721.16 **R squared:** training set 0.02, test set 0.01	**MSE:** training set 713.82, test set 843.29 **R squared:** training set -0.11, test set -0.19	**Best Hyperparameters**: {'max_depth': 10, 'min_samples_leaf': 4, 'min_samples_split': 10, 'n_estimators': 150} **Average MSE** (Cross-Validation): 352.14

NC STATE UNIVERSITY

Methods and analysis

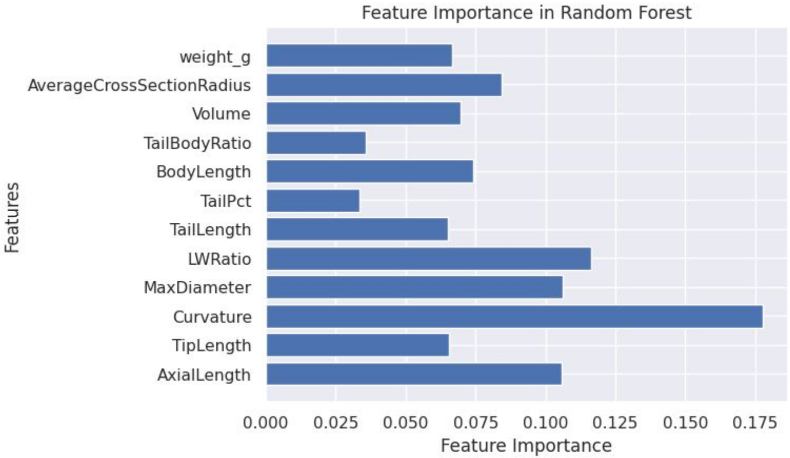
	Lasso	Ridge	Elastic Net	SVR	Random Forest
devUS#1 from grading	**MSE**: training set 299.8, test set 354.0 **R squared**: training set 0.54, test set 0.5	**MSE**: training set 295.24, test set 357.26 **R squared**: training set 0.54, test set 0.5	**MSE:** training set 331.86, test set 386.01 **R squared**: training set 0.54, test set 0.5	**MSE** training set 307.35, test set 367.96 **R squared:** training set 0.52, test set 0.48	**MSE** training : 130.64, test : 286.79 **R-squared** (Training): 0.8, (Test): 0.6
devUS#1 from shape metrics	**MSE**: training set 526.98, test set 623.7 **R squared**: training set 0.37, test set 0.33	**MSE**: training set 429.93, test set 505.76 **R squared**: training set 0.37, test set 0.33	**MSE**: training set 574.3, test set 657.75 **R squared**: training set 0.28, test set 0.22	**MSE** training set 529.59, test set 661.61 **R squared** training set 0.18, test set 0.07	**MSE** training : 173.74, test : 535.82 **R-squared** (Training): 0.73, (Test): 0.25
roundness from shape metrics	**MSE**: training set 350.95, test set 396.87 **R squared:** training set 0.62, test set 0.55	**MSE**: training set 341.05, test set 388.94 **R squared:** training set 0.62, test set 0.55	**MSE**: training set 381.67, test set 418.88 **R squared:** training set 0.62, test set 0.54	**MSE** training set 358.48, test set 409.26 **R squared** training set 0.61, test set 0.53	**MSE** training : 142.0, test : 367.36 **R-squared** (Training): 0.84, (Test): 0.57
tailedness from shape metrics	**MSE**: training set 783.64, test set 781.75 **R squared:** training set 0.33, test set 0.25	**MSE**: training set 728.72, test set 747.71 **R squared:** training set 0.33, test set 0.24	**MSE**: training set 853.13, test set 826.46 **R squared:** training set 0.31, test set 0.25	**MSE** training set 768.38, test set 795.63 **R squared** training set 0.29, test set 0.2	**MSE** training : 260.33, test : 620.9 **R-squared** (Training): 0.76, (Test): 0.38
blockiness from shape metrics	**MSE**: training set 527.44, test set 511.46 **R squared**: training set 0.18, test set 0.16	**MSE**: training set 519.43, test set 518.89 **R squared**: training set 0.2, test set 0.13	**MSE**: training set 538.24, test set 519.33 **R squared**: training set 0.18, test set 0.16	**MSE** training set 541.8, test set 531.25 **R squared** training set 0.16, test set 0.12	**MSE** training : 235.97, test : 509.8 **R-squared** (Training): 0.63, (Test): 0.16
curviness from shape metrics	**MSE:** training set 454.31, test set 591.08 **R squared:** training set 0.12, test set 0.12	**MSE:** training set 439.23, test set 567.62 **R squared:** training set 0.13, test set 0.12	**MSE:** training set 460.03, test set 599.86 **R squared:** training set 0.13, test set 0.12	**MSE** training set 507.88, test set 698.89 **R squared** training set 0.0, test set -0.07	**MSE** training : 97.66, test : 511.14 **R-squared** (Training): 0.81, (Test): 0.21
yvar=['devUS#1'] (Predict from Prediction)	**MSE** training set 599.67, test set 684.79 **R squared** training set 0.08, test set 0.05	**MSE** training set 593.82, test set 674.58 **R squared** training set 0.09, test set 0.06	**MSE** training set 621.14, test set 705.59 **R squared** training set 0.08, test set 0.05	**MSE** training set 675.49, test set 808.96 **R squared** training set -0.05, test set -0.14	**MSE** training : 632.5, test : 709.19 **R-squared** (Training): 0.02, (Test): 0.0

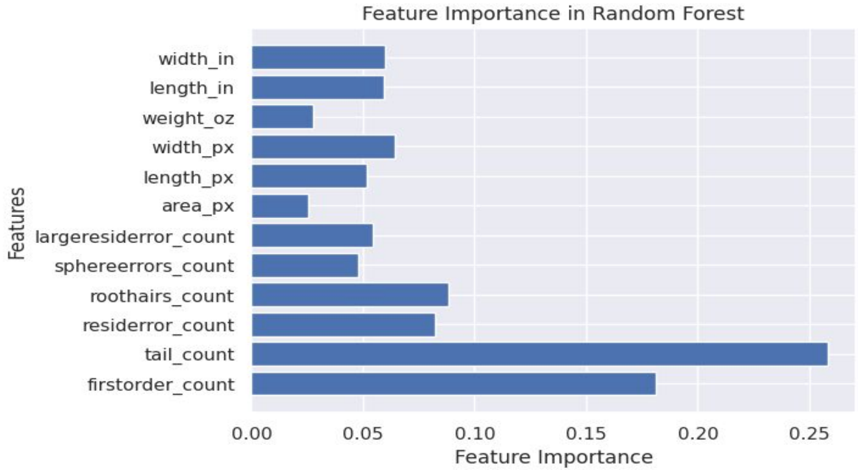
Results

- The Random Forest model has outperformed the other models in terms of performance (MSE and R-square)
- The feature importance obtained for devUS#1 from ***each user-graded metric*** is observed below

Results

- The feature importance obtained for **devUS#1** from **reconstruction** data and **templating** data, respectively, is observed below

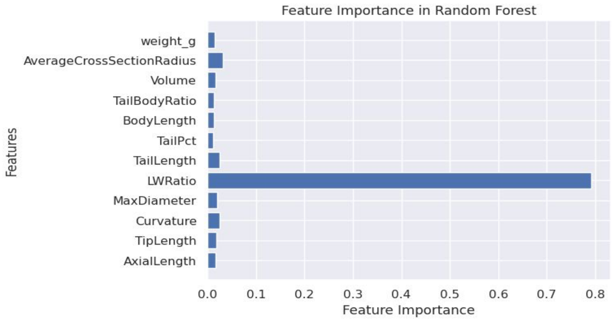




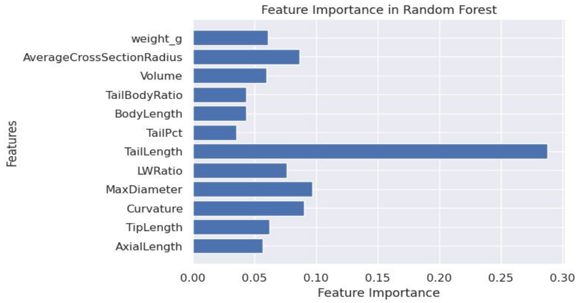
Results

- The feature importance obtained for each **user-graded** metric from **reconstruction** data is observed below

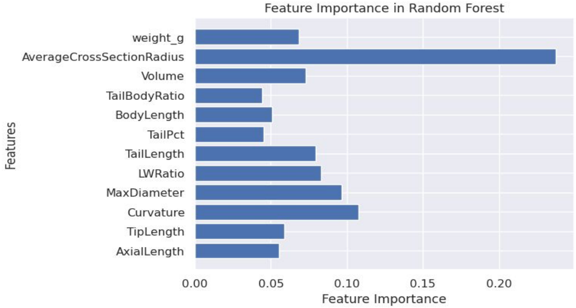
Roundness



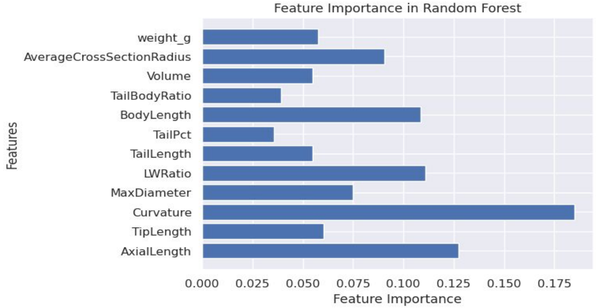
Tailedness



Blockiness



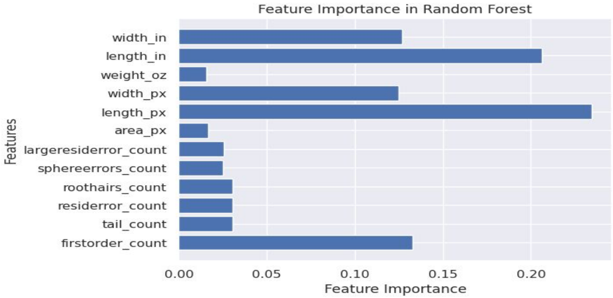
Curviness



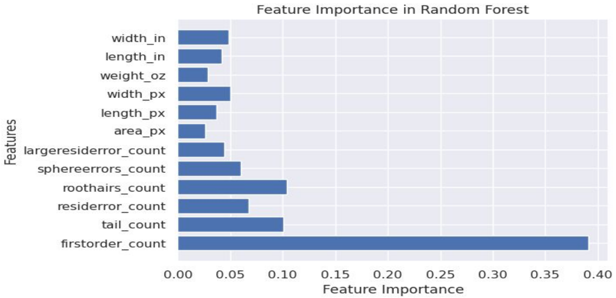
Results

- The feature importance obtained for each **user-graded** metric from **templating/shape metrics** data is observed below

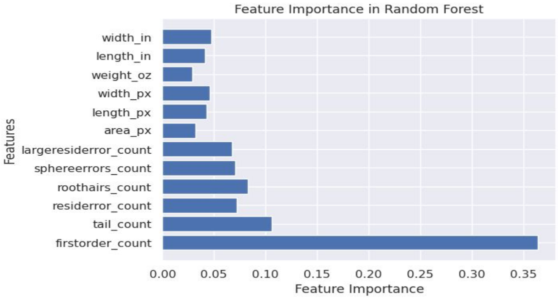
Roundness



Tailedness



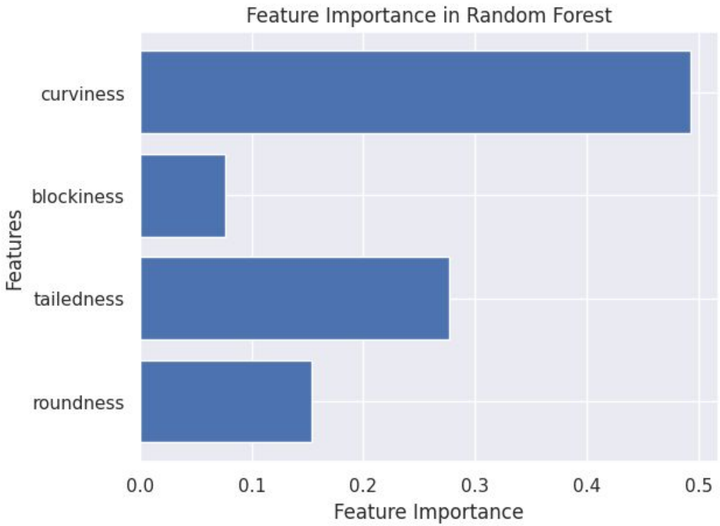
Blockiness

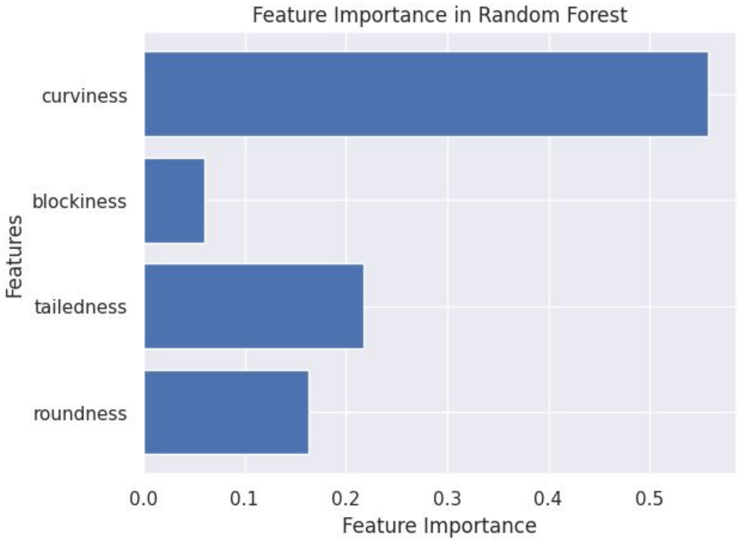


Curviness

Results

Predict from prediction results of **devUS#1** from **reconstruction** data and **templating** data, respectively, is shown below





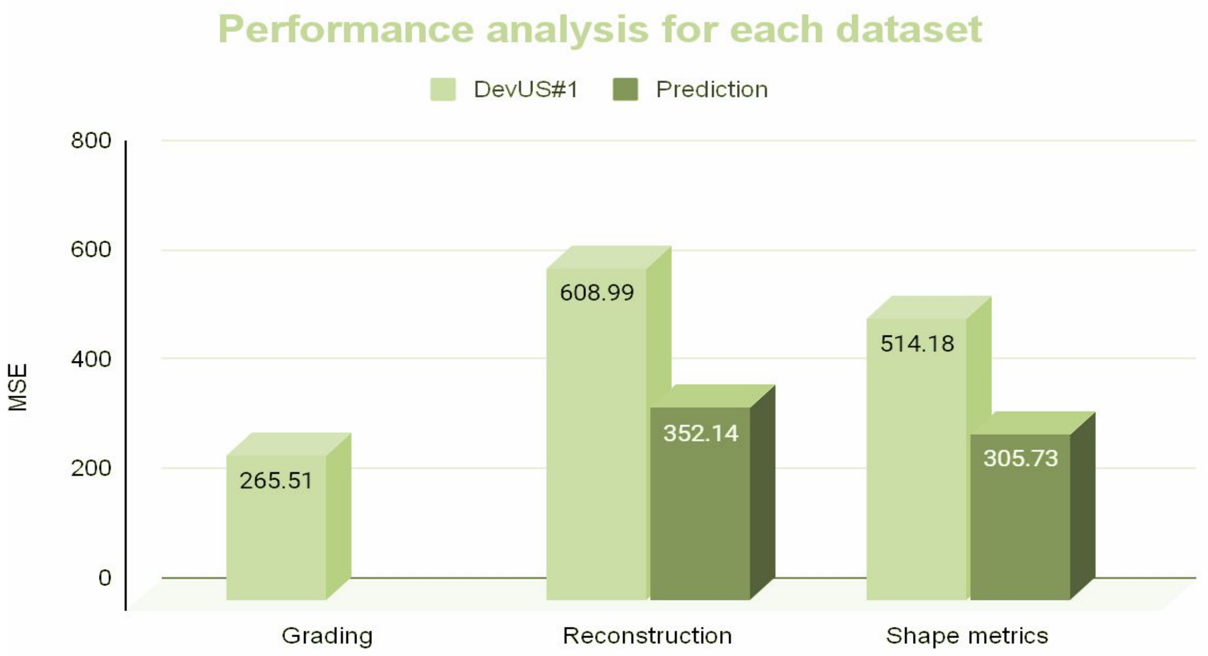
Analysis from results

Our best performing model Random Forest with GridSearchCV and Cross Validation gave the following results:

- MSE for DevUS#1 from Grading: 265.51

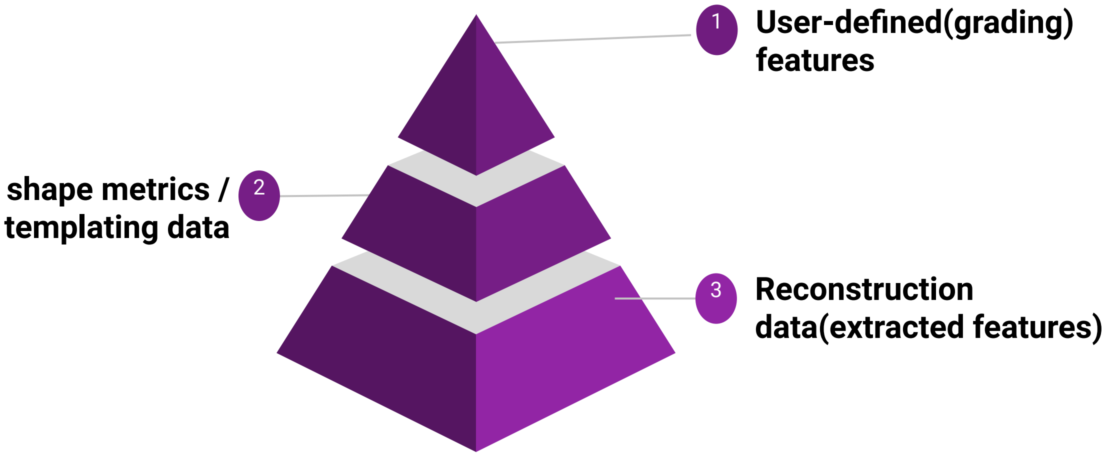
- MSE for DevUS#1 from reconstruction: 608.99
- MSE from prediction for reconstruction from user-defined features: 352.14
(***% improvement in MSE*= 42.17%**)

- MSE for DevUS#1 from shape metrics: 514.18
- MSE from prediction for shape metrics from user-defined features: 305.73
(***% improvement in MSE*= 40.54%**)



Conclusion

After assessing the performance across the three sets of data:



- User-defined(grading) features performed the best in predicting the deviation from US No.1 of a sweet potato and were the most accurate amongst the three.
- The shape metrics/templating data gave better results than the reconstruction data and observed higher improvement in predictions after Random Forest regression.

Future works

- The dataset could be biased where a particular user may not have entered the information in the survey from the GUI sincerely or different users might just have very different perceptions of the idea of US#1 sweet potato. For example, ’tail’ being confused with other error metrics. In such a case, we can modify the dataset instead of or along with a better regression model, such a neural network, to obtain the most optimal results.
- Moreover, we can implement relative scaling on the user-defined data to deal with variation in human scope. In such a case, we would consider the best estimated sweet potato according to the user and scale the rest of the data w.r.t. to that.
- Use of auto-encoders to generate more relevant features of a sweet potato from its image is another possibility we plan to explore in order to obtain better results.

References

1. Samiul Haque, Edgar Lobaton, Natalie Nelson, G. Craig Yencho, Kenneth V. Pecota, Russell Mierop, Michael W. Kudenov, Mike Boyette, Cranos M. Williams. Computer vision approach to characterize size and shape phenotypes of horticultural crops using high-throughput imagery.(https://doi.org/10.1016/j.compag.2021.106011)
2. USDA Agricultural Marketing Service Sweetpotatoes Grades and Standards https://www.ams.usda.gov/grades-standards/sweetpotatoes-grades-and-standards
3. Michael Kudenov’s SweetAPPS Grading via Templating https://lucid.app/lucidspark

THANK YOU!!