REPORT STRUCTURE

1. Abstract

Key Points:

- One or two sentences of background: Kepler KOIs, automated vetting (Robovetter), `koi_score` as a quality/confidence score.
- Project objective: Using physical/observational features from the KOI table, approximate and predict `koi_score` through PCA + four regression models (Linear, Ridge, RF, MLP) to build a lightweight quality assessment surrogate.
- Method overview:
  - Data source and cleaning;
  - Apply PCA on standardized features;
  - Four regression models trained and compared in both Original feature space and PCA feature space.
- Main results (1–2 sentences):
  - Which model + feature combination performs best (e.g., "RandomForest performs best on original features, while linear models are more stable on PCA features");
  - Impact of PCA on performance/training stability.
- One sentence on significance:
  - This model can serve as an auxiliary KOI quality assessment tool, helping to quickly screen high-quality planet candidates.

(Optional) **Index Terms**: Kepler Objects of Interest, koi_score, principal component analysis, regression, quality assessment

---

2. Introduction

> Corresponds to the high-level narrative of 1.1 Motivation & 1.2 Process & 1.3 Conclusion in FRAMEWORK.

Suggested subsections:

2.1 Scientific Background: Kepler KOIs and False Positives

- Brief description of the Kepler mission and KOIs: what are KOIs, sources of transit-like signals.
- Explain false positives: false signal problems caused by binary stars, background stars, noise, etc.
- Introduce automated vetting (Robovetter) and disposition (Candidate / False Positive).

2.2 The Role of `koi_score` as a Quality Metric

- Formally define `koi_score`: a continuous score from 0–1, representing the confidence/quality of vetting results.
- Explain its role in planet statistics/occurrence rate studies (e.g., commonly using threshold 0.9 to screen high-quality samples).
- Emphasize: `koi_score` is the output of a complex pipeline (numerous tests + Monte Carlo injection experiments), which ordinary users cannot easily reproduce.

2.3 Problem Statement and Objectives

- Describe from the perspective of "Advanced Statistical Approaches to Quality":
    - View `koi_score` as a "product/process quality metric," and KOI features as "process characteristics."
- Clearly state the problem definition:
    - Given physical and observational features in the KOI table, can we use PCA + machine learning regression models to approximate and predict the continuous `koi_score`?
- Use 1–2 sentences to explain project objectives:
    - Build a surrogate quality assessment model;
    - Analyze the impact of PCA on model performance, stability, and interpretability;
    - Compare the performance of different regression models in Original vs PCA feature spaces.

2.4 Contributions and Report Organization

- List main contributions (bullets):
  - Build a complete pipeline based on the KOI table for PCA + regression to predict `koi_score`;
  - Systematically compare 4 models (Linear, Ridge, Random Forest, MLP) on original features and PCA features;
  - Analyze the physical meaning of principal components and their relationship with `koi_score`.
- Outline the structure of subsequent sections (corresponding to each section).

---

3. Data Set and Preprocessing

> Corresponds to FRAMEWORK Step 1 (data acquisition, field selection, EDA) + SampleReport's "Data Set Description".

Suggested subsections:

3.1 KOI Cumulative Table and Data Source

- Data source: NASA Exoplanet Archive KOI cumulative table (specify download method/version date).
- Total sample size (original ~9564), actual sample size after cleaning.
- Briefly explain why this dataset is suitable for this course project (real engineering data, multivariate, appropriate size).

3.2 Column Groups and Feature Selection Strategy

- Explain the purpose of each column group according to FRAMEWORK logic:

1. **Identifiers (used only as labels, not in models):**

   `kepid`, `kepoi_name`, `kepler_name`, etc., used for labeling/tracing KOIs in analysis plots.

2. **Target Variable (regression target):**

   `koi_score` (0–1), the only y in this project.

3. **Classification Labels & False Positive Flags (used only for analysis/grouping, not in models):**
    - `koi_disposition` , `koi_pdisposition`
    - `koi_fpflag_nt` , `koi_fpflag_ss` , `koi_fpflag_co` , `koi_fpflag_ec`

      Explanation: They are highly correlated with `koi_score` . To avoid label leakage, they are only used for EDA and results discussion, not for PCA or regression input.
4. **Core Numerical Features for Models (main features of X):**
    - Transit geometry & shape:

      `koi_period` , `koi_time0bk` , `koi_impact` , `koi_duration` , `koi_depth` , `koi_model_snr`
    - Planet properties & irradiation:

      `koi_prad` , `koi_teq` , `koi_insol`
    - Stellar properties:

      `koi_steff` , `koi_slogg` , `koi_srad`
    - Sky position & brightness:

      `ra` , `dec` , `koi_kepmag`
    - Clearly specify pipeline metadata not used as features: `koi_tce_plnt_num` , `koi_tce_delivname` , etc., will be dropped.
5. **Uncertainty Columns (strategy for handling upper/lower error columns):**

    Clearly state: `*_err1` , `*_err2` (such as `koi_period_err1` , etc.) are not retained in this project baseline to avoid dimensional explosion and severe collinearity; only central value columns are used.

3.3 Data Cleaning: Missing Values and Outliers

- Use `df.info ()` , `df.describe()` , etc., to check for missing and anomalous values.
- Explain your strategy for handling missing values:
    - Remove rows missing `koi_score` ;
    - For other features with few missing values, use mean/median imputation;
- Explain handling of extreme outliers (retain/remove/truncate) and provide brief rationale.

3.4 Exploratory Data Analysis (EDA)

- `koi_score` distribution (histogram/KDE): whether skewed, proportion of high/low score regions.
- Histograms and box plots of key features: general range and outlier distribution.
- Correlation matrix heatmap: showing multicollinearity among main features (e.g., whether period, duration, depth, SNR are highly correlated).
- Optional: Plot `koi_score` distribution or distribution in PC space by different `koi_disposition` / false positive flags (will be more detailed in PCA results later).

---

4. Methodology

> Corresponds to the theoretical part of FRAMEWORK Step 2–4 + SampleReport's "PCA" and "Machine Learning-based ...".

Suggested subsections:

4.1 Problem Formulation and Notation

- Mathematical formalization:
  - Given feature vector $\mathbf{x}_i \in \mathbb{R}^p$, the target is continuous $y_i = \text{koi\_score} \in [0, 1]$;
  - Objective: learn mapping $f: \mathbb{R}^p \rightarrow [0, 1]$, minimizing e.g., MSE.
- Briefly state: Subsequently, we will learn 4 different f in both original feature space
  $mathbfx_i$ and PCA feature space $\mathbf{z}_i$.

4.2 Principal Component Analysis (PCA)

- Briefly describe PCA theory (aligned with SampleReport):
  - Standardization, covariance matrix, eigenvalue decomposition;
  - Principal components = linear combinations of original features, sorted by explained variance.
- Specify practical application settings:

    - Perform PCA on standardized numerical features;
    - Does not include target `koi_score` and various flags/labels;
    - Use explained variance ratio with scree plot and cumulative curve to select the top k principal components to retain (e.g., reaching 90–95% cumulative variance).
- Explain the purpose of PCA:
    - Dimensionality reduction;
    - Reduce multicollinearity;
    - Form interpretable "comprehensive physical directions" (e.g., geometry+SNR principal component, stellar scale principal component, etc.).

4.3 Regression Models

- List four regression models and briefly explain their characteristics, referring to FRAMEWORK Step 3 & 4:
    1. **Linear Regression (OLS)** – Baseline linear model, strong interpretability;
    2. **Ridge Regression (L2 regularization)** – Alleviates collinearity, improves generalization;
    3. **RandomForestRegressor** – Nonlinear ensemble tree model, can provide feature importance;
    4. **MLPRegressor (Multi-layer Perceptron)** – Nonlinear neural network, can fit complex relationships.
- Emphasize: Each model will be trained on two feature sets separately:
    - Original standardized features X_{std};
    - PCA features $X_{textpca}$ (top k principal components).

---

5. Experimental Setup

> Consolidates implementation details scattered in SampleReport into one section; also corresponds to FRAMEWORK descriptions of data splitting, standardization, cross-validation, and hyperparameter search.

Suggested subsections:

5.1 Train/Validation/Test Split and Scaling

- Explain split strategy (e.g., 70% / 15% / 15% or 70% / 30%, depending on your actual implementation):
    - Training set for fitting and cross-validation;
    - Validation set for hyperparameter selection (if there is a separate validation set);
    - Test set only for final evaluation.
- Standardization:
    - Use `StandardScaler`;
    - `fit` on training set, apply same transformation to train/val/test;
    - PCA and all models work in standardized space.

5.2 Model Training and Hyperparameter Tuning

- Briefly list key hyperparameters:
    - Ridge's `alpha`;
    - Random Forest's `n_estimators`, `max_depth`, `min_samples_leaf`;
    - MLP's `hidden_layer_sizes`, `alpha`, `learning_rate_init`, `max_iter`, etc.
- Describe hyperparameter selection method:
    - Simple grid search / `RidgeCV` / random search / manual tuning;
    - K-fold cross-validation (specify K value) and scoring metric used (e.g., R^2 or negative MSE).

5.3 Evaluation Metrics

- Clearly list regression evaluation metrics:
    - R^2;
    - MSE, RMSE;
    - MAE;
- Specify which metric is the primary comparison metric (e.g., R^2 as primary, RMSE/MAE as auxiliary).

---

6. PCA Results and Interpretation

> Corresponds to the results section of FRAMEWORK Step 2 + SampleReport's "PCA Results".

Suggested subsections:

6.1 Explained Variance and Choice of k

- Show scree plot and cumulative explained variance curve (similar to SampleReport's Fig. 4 & 5).
- Point out:
  - Variance proportion explained by the first few principal components;
  - Final choice of k (e.g., "the first 6 principal components explain 93% of total variance, so k=6 is selected").

6.2 Principal Component Loadings and Physical Meaning

- Provide summary of loading matrix (can use table/bar chart), showing original features with large weights on each principal component.
- Attempt physical interpretation of the first few principal components:
  - PC1 represents a combination of "strong signal geometry/depth/duration + SNR";
  - PC2 may be related to "stellar scale/temperature";
  - Etc.

6.3 Visualization in PCA Space

- Plot scatter plot of KOIs in (PC1, PC2) plane:
  - Color can use `koi_disposition` or `koi_score` high/low score intervals;
  - Observe whether planet candidates and false positives have structural differences in principal component space.
- Optional: Simple biplot/feature vector arrow diagram to help explain feature projection directions.

---

7. Regression Results

> Change SampleReport's "Classification Results" section to regression results, split into Original vs PCA two parts, corresponding to FRAMEWORK Step 3 & 4 & 5.

Suggested subsections:

7.1 Baseline Performance on Original Features

- Provide table: test set performance of four models on X_{std}:
  - Columns include: Model, R^2, MSE, RMSE, MAE;
  - Can also provide cross-validation mean ± standard deviation.
- Brief analysis:
  - Which models perform best/worst;
  - Whether there are obvious signs of underfitting or overfitting.

7.2 Performance on PCA Features

- Similar to 7.1, provide another table: performance of four models on X_{pca}.
- Comparative analysis:
  - Performance changes of each model on Original vs PCA features;
  - Note whether linear models improve robustness, whether nonlinear models (like RF) change performance due to PCA's "mixed features".

7.3 Error Analysis and Visualizations

- Plots (at least a few):
  - True `koi_score` vs predicted $haty$ scatter plot (preferably for the best model, separately on Original and PCA versions);
  - Residual distribution plot (histogram / QQ-plot / residuals vs predicted values scatter plot).
- Analysis:
  - Error performance in the high `koi_score` (high-quality candidate) range;
  - Whether there is systematic bias (e.g., underestimating high scores).

7.4 Feature Importance and Model Interpretability

- For **RandomForestRegressor (Original features)**:
  - List feature importance ranking, discuss which original features are most sensitive to `koi_score`;
- For **PCA models**:
  - Use PCA loadings + model coefficients (Linear/Ridge) to indirectly interpret the physical meaning behind the most important principal components;
- If time permits, briefly mention SHAP / permutation importance (no need to be as long as SampleReport, can be an additional highlight).

---

8. Discussion

> Corresponds to the comprehensive comparison and interpretation section of FRAMEWORK Step 5.

Should include:

8.1 Summary of Model Comparison

- Synthesize results from 7.1 and 7.2 to answer:
  - Which model + feature combination is "best overall";
  - Whether PCA has positive/negative impact on overall performance and model stability;
  - Trade-offs between different models (interpretability vs performance vs training cost).

8.2 Implications for Quality Assessment of KOIs

- Discuss from a quality engineering perspective:
  - Is the model more accurate in the high `koi_score` range? (e.g., reliability for truly high-quality candidates);
  - Can model output be used for rapid quality screening (e.g., first eliminate obviously low-score KOIs, then use more expensive vetting pipeline).

8.3 Limitations and Threats to Validity

- `koi_score` itself contains noise and may change with pipeline updates;
- Unused complex features (e.g., pixel-level data, time series information);
- Model may be affected by training sample distribution bias;
- PCA mixes features, which may weaken some astronomical interpretability.

---

9. Conclusion and Future Work

> Corresponds to "Conclusion: Report and Deliverables" in FRAMEWORK 1.3.

Should include:

9.1 Conclusion

- Review:
  - Motivation for selecting KOI dataset and `koi_score`;
  - Overall workflow of EDA + PCA + four regression models + Original vs PCA comparison;
- Summarize main findings:
  - Predictability of `koi_score` (approximate R^2 range);
  - Most important features/principal components;
  - Role of PCA/nonlinear models in quality assessment.

9.2 Future Work

- Possible extension directions:
  - Add more features (including false positive flags, error columns, more stellar/planet parameters);
  - Try stronger models (Gradient Boosting, XGBoost, Gaussian Process, etc.);
  - Extend `koi_score` regression to classification tasks (high/low quality KOI classification, terrestrial planet screening, etc.);
  - Introduce time series features or Bayesian modeling to improve uncertainty estimation.

---

10. References

- Papers & documentation:
    - Kepler mission, Robovetter, `koi_score` official documentation and papers;
    - NASA Exoplanet Archive documentation;
    - PCA/regression/MLP related textbooks or papers;
    - If interpretability analysis (SHAP, etc.) is included, cite corresponding papers/books.
- Organize according to IEEE or your chosen citation format.