# COMS 4995 Applied Machine Learning | Assignment 1

*Luca* Barattini – UNI: LB3656[1,⋆]

[1] *Columbia University in the City of New York, Fu Foundation School of Engineering*

**Abstract.** This report documents an end-to-end machine learning workflow for predicting survival on the RMS Titanic using the labeled Kaggle dataset. The pipeline covers cleaning, imputation, exploratory data analysis, feature engineering, model training, and comparative evaluation. Two modeling families were studied during the research: a generative approach based on Naive Bayes, and a discriminative approach based on Linear Regression with Ridge and LASSO regularisation. All models were trained on the same stratified train-test validation split, with cross validation used for stability and hyperparameter search. Evaluation relies on accuracy, precision, recall, F1, AUC, confusion matrices, and ROC curves.

## 1 Introduction

The sinking of the Titanic ocean liner is among the most studied maritime tragedies. On 15 April 1912, during her maiden voyage, the Titanic struck an iceberg and went down in the North Atlantic. Of the 2224 people on board, 1502 tragically passed away. Survival was not purely random. Contemporary accounts and historical analyses agree that women, children, and first class passengers had better chances of reaching a lifeboat.

This project uses the training data from the Kaggle Titanic challenge to build a complete flow from raw data to predictive models. The objective is pragmatic. We want a clean and reproducible machine learning pipeline that ingests imperfect data, manipulates and enriches it, trains both generative and discriminative models, and evaluates them with clear metrics and visualizations.

Concretely, we will:

- Clean and transform the raw passenger records, addressing missingness and noisy inputs
- Engineer features that reflect socio demographic structure, including family composition, titles, and fare related transformations
- Train and compare two modelling families: a generative Naive Bayes view and a discriminative linear regression view with regularization
- Evaluate with accuracy, precision, recall, F1, and AUC, supported by confusion matrices and ROC curves
- Reflect on the workflow, including the selection of hyperparameters for Ridge and Lasso and the effect of smoothing in Naive Bayes

⋆e-mail: lb3656@columbia.edu

The outcome is not just an evaluation matrix. It is a compact case study in how careful preprocessing and targeted features make simple models competitive and interpretable.

## 2 Data cleaning and imputation

The dataset shows three possible issues: missing values, a few inconsistencies, and heavy skewness in some numerical variables.

### 2.1 Missingness

- **Cabin:** Missing for roughly seventy percent of the dataframe, therefore, we drop the column rather than imputing letters. Figure 1 shows the imbalance.

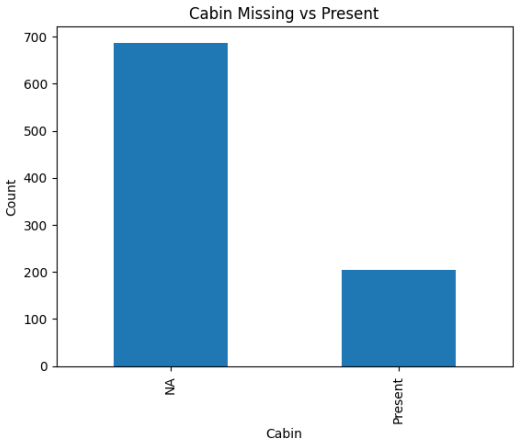



**Figure 1.** Cabin recorded vs missing.

- **Embarked:** Two rows are missing. Within each passenger class, the most common port is stable, so we impute by the class-specific mode.
- **Age:** Moderately missing. We extract titles from names and impute the median age within each title by class

group. Figure 2 shows that the imputed distribution well aligns with the original one.

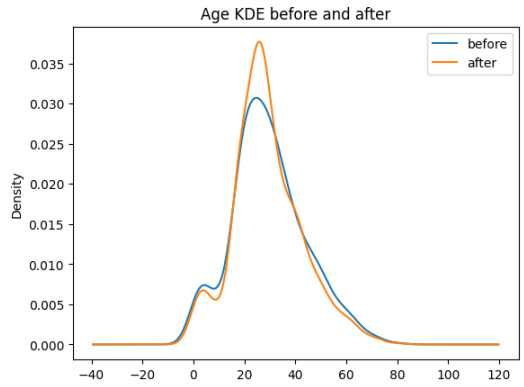



**Figure 2.** Age distribution before and after imputation (KDE).

### 2.2 Consistency and outliers

Passenger identifiers are monotonic with no duplicates; ticket strings contain only plausible characters.

Boxplots for age, fare, siblings/spouses, and parents/children reveal long right tails but nothing obviously invalid (Fig. 3). Extreme fares belong to first class and are historically plausible; therefore, we retain them.

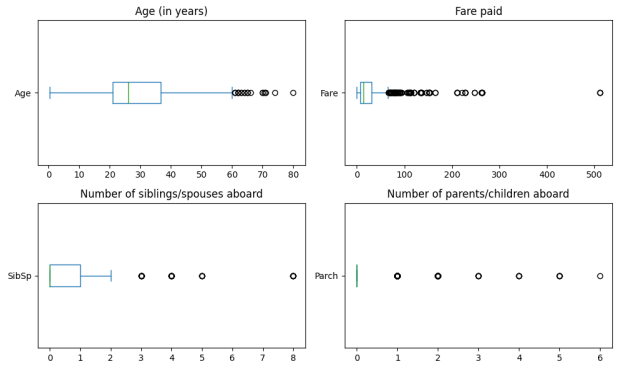



**Figure 3.** Boxplots for age, fare, siblings/spouses, and parents/children.

After these steps the frame is free of missing values and major inconsistencies.

## 3 Exploratory data analysis

### 3.1 Univariate analysis

**Target balance:** The training set contains 891 passengers: about 38% survived and 62% did not. The class imbalance in Fig. 4 motivates looking beyond accuracy alone.

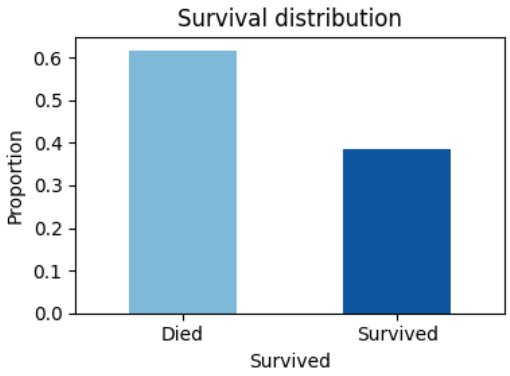



**Figure 4.** Distribution of the target variable (survived vs died).

**Demographics.** Class composition and basic counts are summarised by Figs. 3 and 5. Third class accounts for more than half of the sample, with first and second forming the remainder (Fig. 5). The age panel in Fig. 3 is right-skewed, centred in the late twenties, with a thin tail into older ages. The siblings/spouses and parents/children panels show heavy mass at zero with sparse long tails, indicating many solo travellers or small parties and few large family groups. These shapes anticipate the usefulness of features such as family size and an indicator for travelling alone.

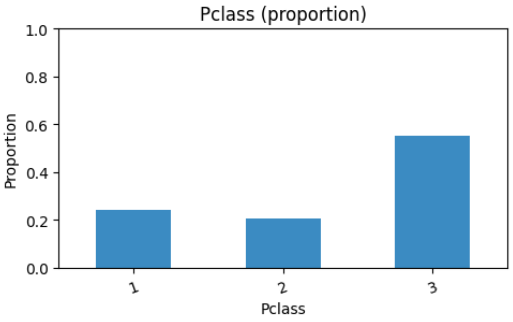



**Figure 5.** Proportion of passenger classes.

**Fares.** Ticket prices are highly right-skewed with a small number of very expensive tickets. As shown in Fig. 6, these concentrate in first class, so we treat them as signal rather than error. Downstream we stabilise fares by normalising per family size, relativising by the class median, and applying a log transformation.

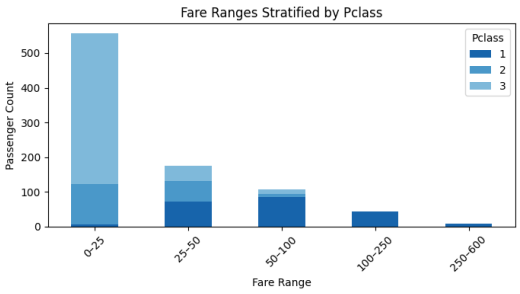



**Figure 6.** Fare ranges stratified by passenger class.

### 3.2 Multivariate patterns

**Correlations among numerics:** The absolute correlation heatmap (Fig. 7) gives a compact overview. Siblings/spouses and parents/children well correlate with

family size by construction. Fare shows moderate association with survival, consistent with socio-economic advantage. Age has weaker direct correlation but becomes informative once stratified.

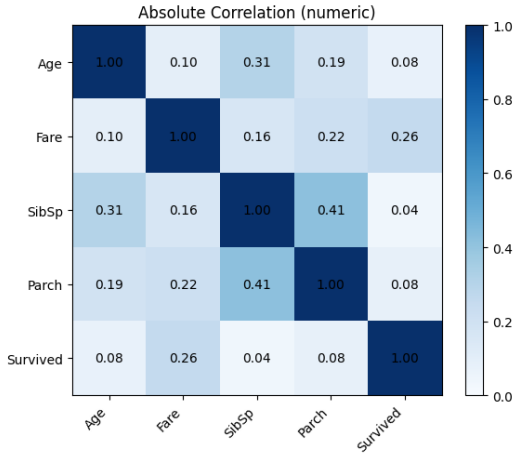



**Figure 7.** Absolute correlation heatmap for key numeric variables.

**Crosstabs and survival rates:** Stacked bars and heatmaps separate high and low risk groups: women survive at far higher rates than men; first class sits well above second and third; children survive more than adults. The 2D view in Fig. 9 shows females in first or second class with very high survival probability, while males in third class rarely survive. This structural interaction motivates the sex-by-class features used later.

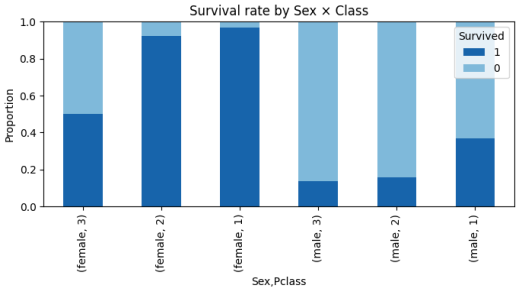



**Figure 8.** Stacked survival rates by sex and class.

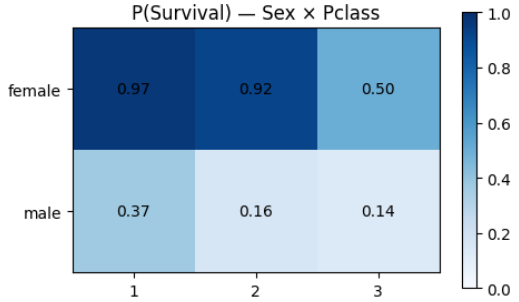



**Figure 9.** Heatmap: probability of survival by sex and class.

## 4 Feature engineering

The aim is to turn the qualitative patterns uncovered in exploration into compact, model-friendly signals. We privilege features that (i) summarise social structure (household composition, status cues from names), (ii) stabilise skewed numerics while preserving rank information (fare transformations), and (iii) make dominant interactions explicit so that even simple linear models can exploit them.

### 4.1 Family and companionship

- **FamilySize** equals SibSp plus Parch plus one for self.
- **IsAlone** flags FamilySize equal to one.
- **TicketGroupSize** counts passengers sharing a ticket number; **TicketShared** flags groups larger than one.

### 4.2 Demographic refinements

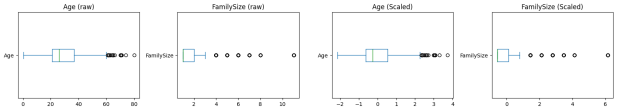
- **Title** is extracted from the name and consolidated: noble forms merge under *Royalty*; military and professional forms become *Officer*; variants of *Miss* and *Mrs* are harmonised.
- **Mother** indicates an adult female with at least one child on board and a title other than *Miss*.
- **IsChild** marks age below 16.
- **AgeBin** slices age into ordered bins and one-hot encodes them. In Gaussian and linear views we keep a continuous age plus an indicator flagging imputed ages.

### 4.3 Socio-economic signal

- **FarePerPerson** equals Fare divided by FamilySize to normalise group purchases.
- **Fare_rel_Pclass** divides FarePerPerson by the median FarePerPerson within each passenger class; we then apply a log transform and standardise to zero-mean, unit-variance. This relative, stabilised measure carries price information net of class effects.


(a) Before log transform (b) After log transform

**Figure 10.** Fare distribution before and after log scaling: the transform reduces right-skew while preserving rank information.




(a) Before log transform (b) After log transform

**Figure 11.** Class-relative fare (Fare_rel_Pclass) before and after log scaling: tighter, more symmetric spread that is easier to model.

## 5 Model training

Our aim is to create an apples-to-apples comparison across model families on a single evaluation set. Three design choices ensure fairness: (i) one stratified split is created once and reused across all views; (ii) each algorithm receives a feature view tailored to its assumptions while keeping information content comparable; (iii) all models

output probabilities clipped to $[0, 1]$ and are thresholded at 0.5 so confusion matrices and AUC are directly comparable. We then append evaluation metrics on a results dataframe.

### Train/test split reused across views

```
idx = np.arange(len(df))
itr, ite = train_test_split(idx, test_size=.30,
stratify=df["Survived"], random_state=42)
```

## 5.1 Naive Bayes

### *5.1.1 BernoulliNB*

BernoulliNB models a vector of independent Bernoulli trials per class. This matches a design where many predictors are indicators: presence of a title, membership in a family-size bucket, ticket-shared flag, age-bin dummies, and fare placed into quantile bins. In this space, each coefficient answers how often a trait appears among survivors versus non-survivors under conditional independence.

Laplace smoothing prevents zero likelihoods for rare indicators. Without it, a single unseen combination can collapse the posterior. We therefore probe small but meaningful values of the smoothing parameter $\alpha$ and report two representative settings. On the binarised view we fit BernoulliNB at $\alpha = 0.1$ and $\alpha = 0.5$. Operating characteristics are essentially identical: confusion counts change marginally and AUC is stable.

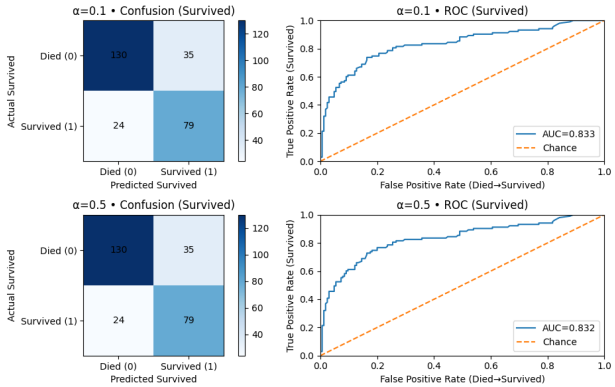



**Figure 12.** BernoulliNB at $\alpha = 0.1$ and $\alpha = 0.5$: confusion matrices and ROC.

**Table 1.** BernoulliNB on the held-out set.

| $\alpha$ | Acc | Prec | Rec | F1 | AUC |
|---|---|---|---|---|---|
| 0.1 | 0.780 | 0.693 | 0.767 | 0.728 | 0.833 |
| 0.5 | 0.780 | 0.693 | 0.767 | 0.728 | 0.832 |

To check sensitivity, we sweep $\alpha \in \{0, 10^{-6}, 0.1, 0.5\}$. ROC curves almost overlap, which signals that smoothing primarily stabilises the tails for very rare indicators without altering the decision boundary dominated by common traits.

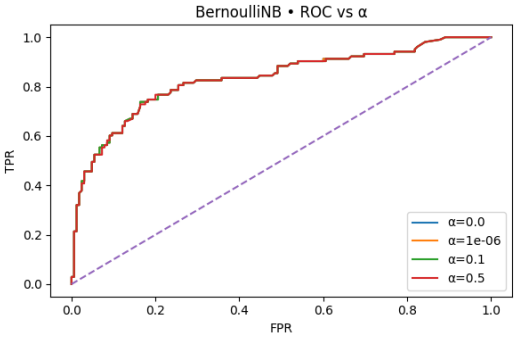



**Figure 13.** BernoulliNB ROC across $\alpha \in \{0, 10^{-6}, 0.1, 0.5\}$: curves largely overlap.

### *5.1.2 GaussianNB*

GaussianNB assumes each feature is conditionally normal within class and independent given the class. This view preserves continuous signals that exploration showed to be informative after stabilisation, such as age and the log-standardised fare relative to class. The variance-smoothing parameter adds a small value to feature variances to avoid vanishing denominators and to regularise noisy variance estimates when sample sizes are modest. On the continuous view, GaussianNB attains a higher AUC than the Bernoulli variants. The confusion matrix shows a balanced error profile.

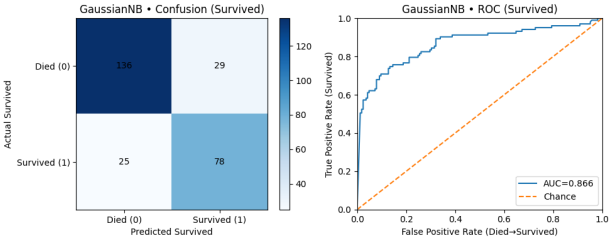



**Figure 14.** GaussianNB: confusion matrix and ROC on the continuous view.

**Table 2.** GaussianNB on the held-out set.

| Model | Acc | Prec | Rec | F1 | AUC |
|---|---|---|---|---|---|
| GaussianNB | 0.799 | 0.729 | 0.757 | 0.743 | 0.866 |

We sweep the variance-smoothing parameter across orders of magnitude. ROC curves change gently, indicating that class-conditional density estimates are already stable and the smoothing acts as a safety margin rather than a tuning knob.

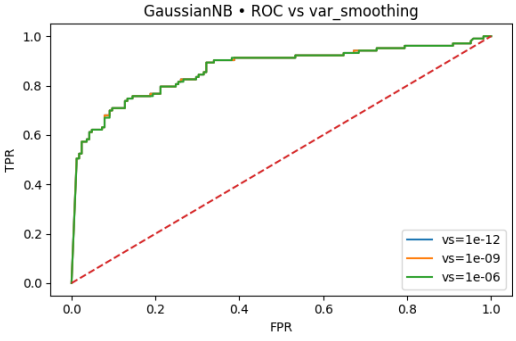



**Figure 15.** GaussianNB ROC under several variance-smoothing values: small differences across decades.

### 5.2 Linear regression family

Although a logistic link is standard for binary outcomes, here we use linear regression as a baseline classifier. We fit an ordinary least squares model and clip its predictions to $[0, 1]$, then classify them with a fixed 0.5 threshold. We subsequently fit Ridge and Lasso in identical preprocessing pipelines so that any performance gains can be ascribed to regularisation rather than to feature handling.

#### *5.2.1 Scaling and pipeline*

Continuous variables are scaled using StandardScaler, while all other columns pass through unchanged. Each model is wrapped in a scikit-learn Pipeline so that preprocessing is learned on the training fold and applied consistently to the hold-out set.

#### *5.2.2 Choosing $\alpha$ for Ridge and Lasso*

We select the regularisation strength with five-fold StratifiedKFold cross-validation and a custom scorer that measures F1 at the 0.5 threshold. Candidates are drawn from a dense log-spaced grid. GridSearchCV is run once with Ridge as the final step and once with Lasso (with a high max_iter). The best $\alpha$ from cross-validation is then refit on the full training split and evaluated on the held-out slice, aligning model selection with the classification objective rather than a regression loss.

#### *5.2.3 Confusion matrices and ROC*

Across the three members of the linear family, ROC profiles are very similar, with AUC around 0.88 on this split. Confusion matrices show high true negatives and solid true positives, reflecting a balanced trade-off.

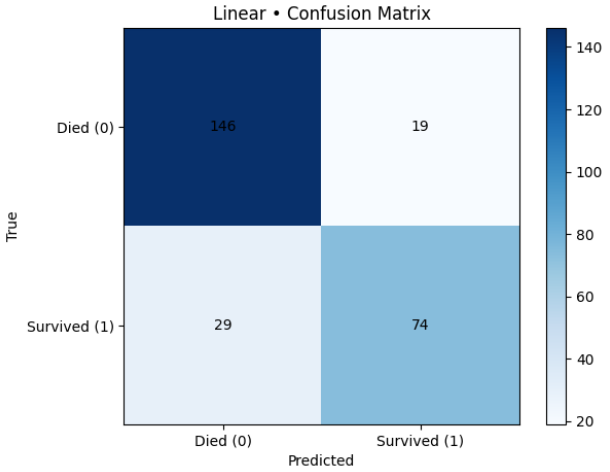



**Figure 16.** Linear (OLS) baseline: confusion matrix on the scaled view.

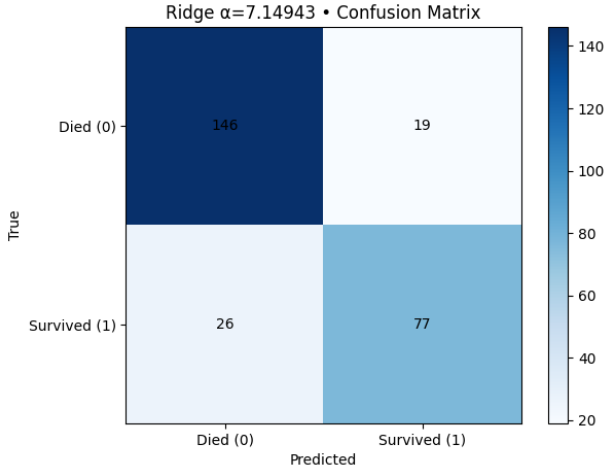



**Figure 17.** Ridge (best $\alpha$ from CV): confusion matrix.

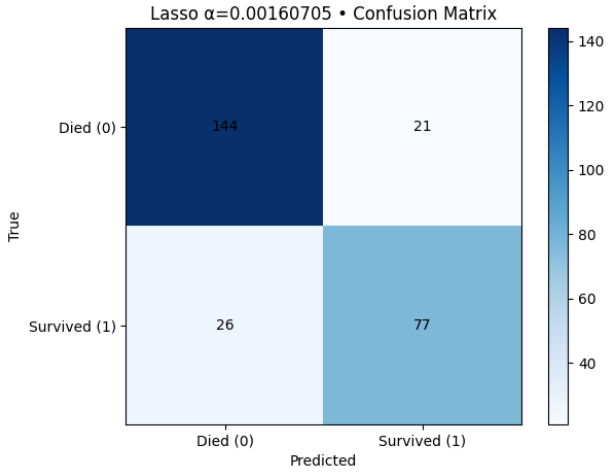



**Figure 18.** Lasso (best $\alpha$ from CV): confusion matrix.

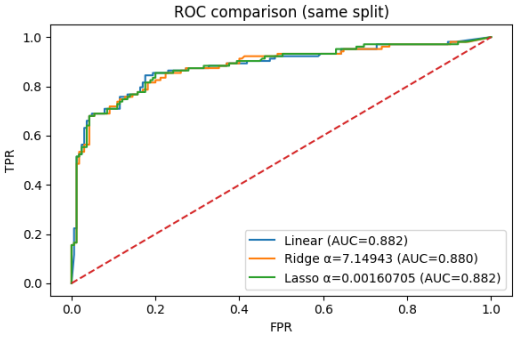



**Figure 19.** ROC comparison across the linear family (OLS, Ridge, Lasso) on the held-out set.

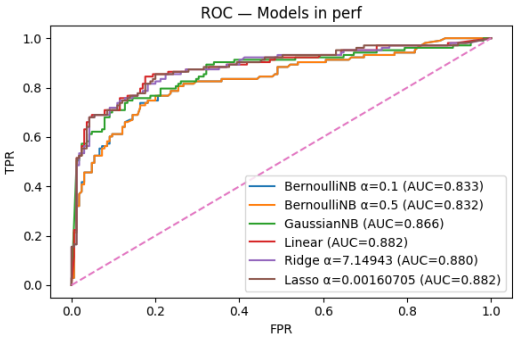



**Figure 20.** ROC comparison across all fitted models (BernoulliNB, GaussianNB, OLS, Ridge, Lasso).

# 6 Model comparison and evaluation

## 6.1 Metrics across models

We collect predictions and hard labels for every model on the same test indices, then compute accuracy, precision, recall, F1, and AUC.

The ranking is stable across metrics:

- The three linear models form the top group with AUC near 0.88.
- GaussianNB follows with AUC around 0.87.
- BernoulliNB is a solid but weaker baseline with AUC near 0.83.

Precision is highest for the linear family, which means that when a survival is predicted it is usually correct. Recall differs a little because the threshold is fixed at 0.5 for all models.

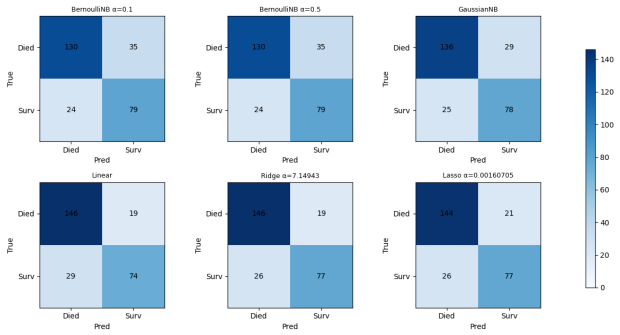



**Figure 21.** Confusion matrices for all fitted models on the common test split.

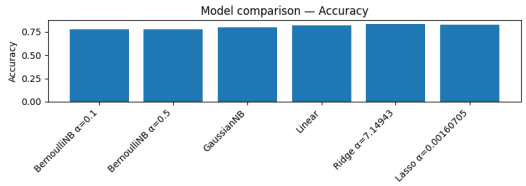



**Figure 22.** Accuracy across models.

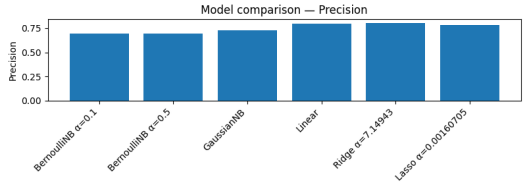



**Figure 23.** Precision across models.

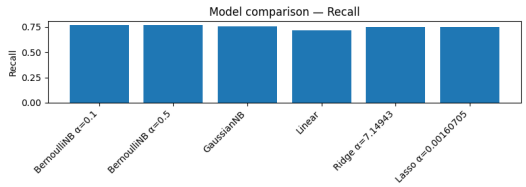



**Figure 24.** Recall across models.

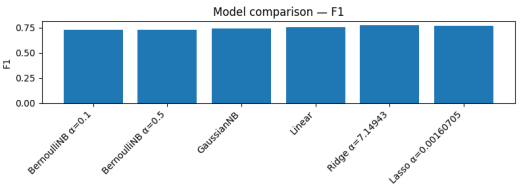



**Figure 25.** F1 across models.

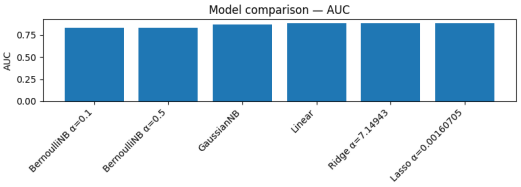



**Figure 26.** AUC across models.

**Table 3.** Held-out performance on the common test split.

| Model | Acc. | Prec. | Rec. | F1 | AUC |
|---|---|---|---|---|---|
| BernoulliNB $\alpha$=0.1 | 0.780 | 0.693 | 0.767 | 0.728 | 0.833 |
| BernoulliNB $\alpha$=0.5 | 0.780 | 0.693 | 0.767 | 0.728 | 0.832 |
| GaussianNB | 0.799 | 0.729 | 0.757 | 0.743 | 0.866 |
| Linear | 0.821 | 0.796 | 0.718 | 0.755 | 0.882 |
| Ridge $\alpha$=7.14943 | 0.832 | 0.802 | 0.748 | 0.774 | 0.880 |
| Lasso $\alpha$=0.00160705 | 0.825 | 0.786 | 0.748 | 0.766 | 0.882 |

Table 3 summarises the numbers behind the plots: the linear family clusters at the top (AUC $\approx$ 0.88), GaussianNB follows ($\approx$ 0.87), and BernoulliNB trails ($\approx$ 0.83).

## 6.2 What the confusion matrices say

The matrices in Figure 21 provide a concrete count level view.

- **Linear baseline:** high true negatives, good number of true positives, few false positives.
- **Ridge:** very close to the linear baseline, as expected with mild shrinkage.
- **Lasso:** similar balance of errors, with a few low value indicators pruned by sparsity.
- **GaussianNB:** slightly more false positives than the linear family yet strong separation.
- **BernoulliNB:** more conservative and a bit less sensitive, aligning with its lower AUC.

This matches the bar charts: linear models trace the top band, GaussianNB is competitive, and BernoulliNB trails slightly.

## 6.3 Why the ranking makes sense

The engineered features are correlated by design. Title is related to sex and age. The relative-fare measure correlates with class and port. SexClass explicitly encodes their interaction. Linear models do not assume independence and therefore can leverage these correlations directly; GaussianNB still benefits from continuous per-class summaries; BernoulliNB loses within-bin variation.

## 7 Interpretation and ablations

### 7.1 Feature effects in the linear view

Coefficient signs accord with historical expectations: being male has a strong negative association with survival. Title indicators for Mrs and Miss are positive relative to the reference, consistent with lifeboat priority. Being in lower classes carries negative weight, while higher relative fares carry positive weight. Child and Mother indicators are positive. Sex by class indicators for males in lower classes carry negative weight, while the female combinations carry positive weight.

### 7.2 LASSO sparsity audit

With $\alpha = 0.00160705$, Lasso trims predictors whose marginal contribution is negligible given the rest of the design. In our run, the following coefficients were driven numerically to zero ($|w_i| < 10^{-8}$):

**Table 4.** Features set to zero by LASSO.

| Feature name |
|---|
| remainder__Mother |
| remainder__Title_Royalty |
| remainder__SC_F2 |
| remainder__SC_M1 |

*Why these likely went to zero?*

- **Mother** is largely explained by the combination of age, title (e.g., *Mrs*), and child/parent counts; once those are present, its extra signal is redundant.
- **Title_Royalty** is extremely rare; LASSO prefers to spend degrees of freedom on common, higher-signal indicators.
- **SC_F2** and **SC_M1** (sex×class dummies) are partly captured by other interactions and the stabilised fare signal; the strongest contrasts (e.g., male third class, female first class) remain, while mid-level contrasts can be dropped with little loss.

This sparsity is useful: it highlights a minimal subset that still delivers the performance reported in Table 3. Different $\alpha$ values or small resplits can alter which near-zero terms get pruned, but the qualitative picture is stable.

### 7.3 Smoothing and Naive Bayes

The Bernoulli ROC curves for multiple alphas are nearly indistinguishable. Smoothing primarily prevents zero probabilities for rare dummy and class combinations. On this dataset, where one hot features already isolate strong patterns such as sex, class, and title, the posterior is driven by common features whose likelihoods are well estimated even without heavy smoothing. GaussianNB shows mild sensitivity to variance-smoothing, as expected, but performance remains stable across a reasonable range.

## 8 Conclusion

This project built a clear path from raw records to working models for survival prediction on the Titanic. The data required deliberate cleaning. Cabin was mostly missing and was dropped. Embarked had two missing entries and was imputed by class specific mode. Age was imputed using medians within title by class groups, which preserved structure and avoided distortion. Sanity checks confirmed that identifiers and categoricals behaved as expected. Outlier analysis indicated that extreme fares belong to first class and should remain.

Feature engineering then translated exploration into compact signals. Family composition, titles, motherhood, childhood, ticket sharing, and a fare measure relative to class were all included. One hot encodings and an interaction between sex and class gave linear models access to the strongest non additivity in the data. Views were tailored per model family: a binarised set for BernoulliNB, a continuous set for GaussianNB, and a scaled set for the linear models.

On a common test slice, the discriminative linear family achieved the best ranking ability with AUC near 0.88, while GaussianNB followed closely and BernoulliNB provided a solid baseline at about 0.83 AUC. Confusion matrices and metric bars tell a consistent story. Hyperparameters for Ridge and Lasso were selected by five fold cross validation on an F1 at 0.5 scorer over a wide log spaced grid, yielding stable, well calibrated models. Lasso sparsity made interpretation easy by excluding low value indicators.

The broader lesson is methodological. With careful preprocessing and light but thoughtful features, simple, interpretable models perform strongly. Correlations and interactions among socio demographic variables are the core drivers of outcome here. Modelling choices that respect that structure, even in straightforward forms, deliver both accuracy and insight.

## 9 AI tool usage disclosure

I used ChatGPT to speed up technical editing, check plotting syntax, and pressure test explanations. Code and text were always inspected, verified, and adjusted by me. The analytical decisions, feature definitions, and model comparisons reflect my work on the dataset.

## 10 Appendix

**Table 5.** Features zerod by LASSO.

| Feature name |
|---|
| remainder__Mother |
| remainder__Title_Royalty |
| remainder__SC_F2 |
| remainder__SC_M1 |

**Table 6.** Lasso top 15 coefficients by magnitude

| feature | coef |
|---|---|
| $\text{remainder}_{Title\_Mrs}$ | 0.1704 |
| $\text{remainder}_{Title\_Miss}$ | 0.1082 |
| $\text{remainder}_{Emb\_Q}$ | 0.0639 |
| $\text{remainder}_{Emb\_C}$ | 0.0457 |
| $\text{remainder}_{TicketShared}$ | 0.0301 |
| $\text{remainder}_{FarePerPclass}$ | 0.0299 |
| $\text{remainder}_{IsChild}$ | 0.0270 |
| $\text{remainder}_{Age\_missing}$ | -0.0199 |
| $\text{scale}_{Age}$ | -0.0668 |
| $\text{scale}_{FamilySize}$ | -0.0782 |
| $\text{remainder}_{Title\_Officer}$ | -0.1609 |
| $\text{remainder}_{SC\_M2}$ | -0.2573 |
| $\text{remainder}_{SC\_M3}$ | -0.2824 |
| $\text{remainder}_{SC\_F3}$ | -0.3773 |
| $\text{remainder}_{Title\_Mr}$ | -0.3960 |