Business Data Analytics – Lecture 6: Regression for Business (OLS, Regularization, Importance, SHAP)

Assoc. Prof. Nguyen Binh Minh, Ph.D.

Learning Objectives (CLO)

- Translate business questions into regression problems.
- Fit and diagnose Ordinary Least Squares (OLS); understand key assumptions.
- Apply regularization (Ridge/Lasso) to handle noise & collinearity.
- Explain models via feature importance and SHAP-like contributions.

Agenda

1) Problem framing & data
2) Ordinary Least Squares (OLS) fundamentals & diagnostics
3) Multicollinearity & transformations
4) Regularization: Ridge, Lasso, Elastic Net
5) Feature importance: coeffs, permutation
6) SHAP concepts & linear contributions
7) Case study & hands-on; quiz

Running Case — Monthly Revenue

Synthetic telco-like dataset (n≈1.4k).

Target: Monthly Revenue; mix of numeric & categorical features.

Business levers: usage, pricing, plan type, complaints.

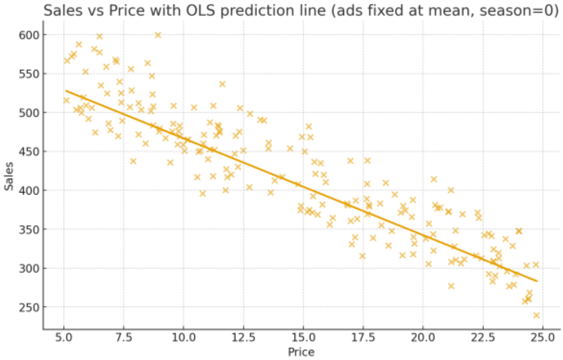
OLS — Intuition

- Model: $y = X\beta + \varepsilon$
  Goal: $\min_{\beta} \ \| y - X\beta \|^2$
  Predict $\hat{y} = X\hat{\beta}$, phần dư $e = y - \hat{y}$,
  variance estimate $\hat{\sigma}^2 = \| e \|^2/(n - p)$.
- Find coefficients minimizing squared errors.
- Captures linear effects and interactions (if added).
- Interpretable under proper scaling and coding.

OLS — Assumptions

- Linearity; no perfect multicollinearity.
- Exogeneity (errors uncorrelated with predictors).
- Homoscedastic & i.d. errors; normality for inference.

Scatter Sales vs Price with OLS forecast line

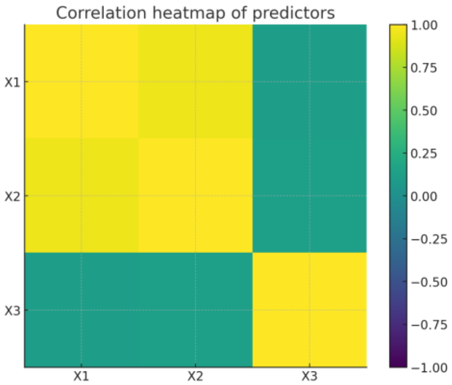


OLS — Estimation

- Closed form: $\beta = (X^T X)^{-1} X^T y$ (or pseudo-inverse).
- Standard errors from residual variance and $(X^T X)^{-1}$.

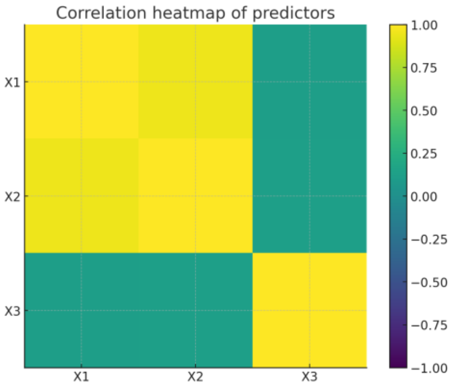
Multicollinearity — What & Why

- Strongly correlated predictors inflate variance.
- Unstable coefficients; wide intervals.



Collinearity Checks

- Correlation heatmap.
- Drop/merge features or regularize.

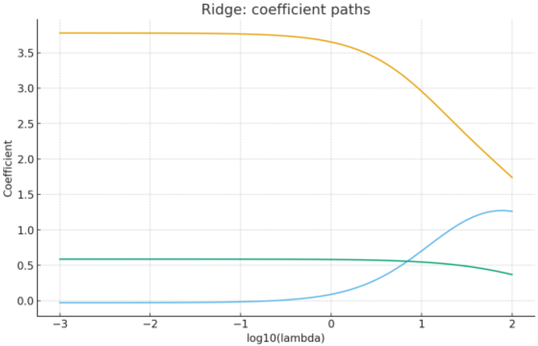


Regularization — Big Picture

- Bias–variance trade-off: shrink coefficients to reduce variance.
- Better generalization; handles noisy & collinear features.

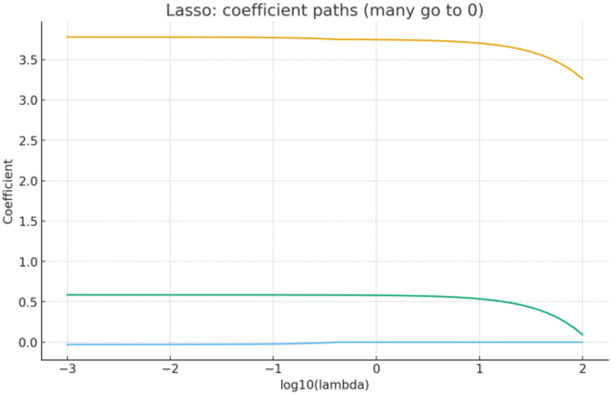
Ridge (L2)

- Penalty $\lambda \sum \beta_j^2$; closed-form solution; never zeros coefficients.
- Good for multicollinearity; smooth shrinkage paths.



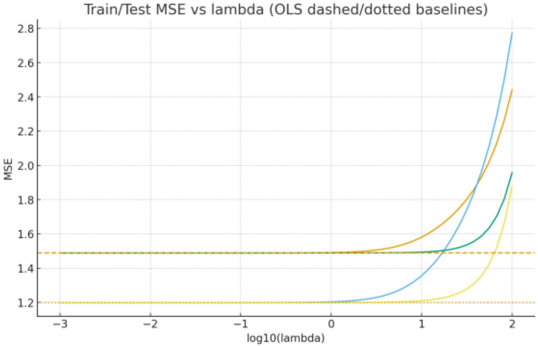
Lasso (L1)

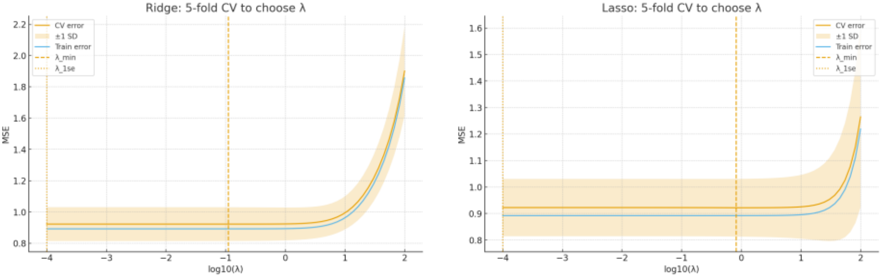
- Penalty $\lambda\sum|\beta_j|$; induces sparsity (feature selection).
- No closed form; solved by coordinate descent.



Elastic Net ($\alpha \cdot L1 + (1-\alpha) \cdot L2$)

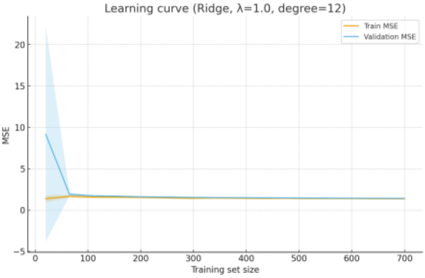
- Combines selection (L1) and stability (L2).
- Useful with correlated groups of features.

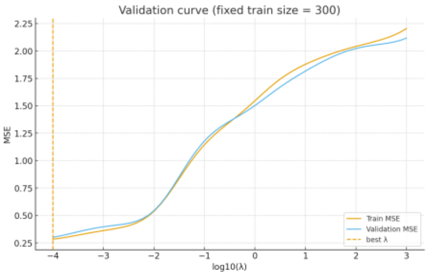




Choosing λ (Regularization Strength)

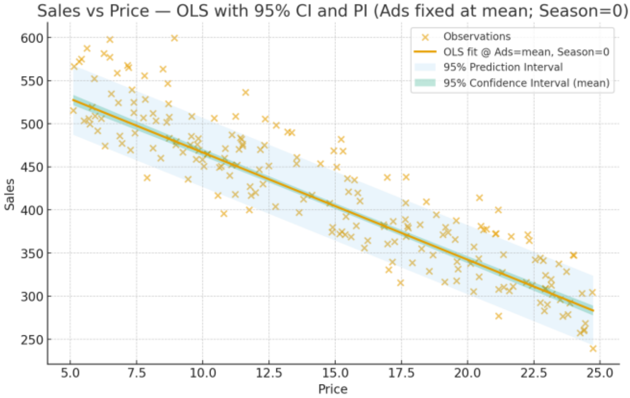
- Cross-validation over a grid of λ.
- Watch for data leakage; standardize features.





Model Selection & Evaluation

- Train/valid/test split; K-fold CV.
- Metrics: RMSE/MAE; adjusted R^2 for comparability.

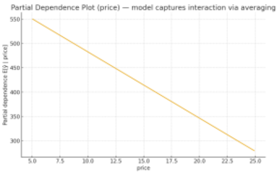


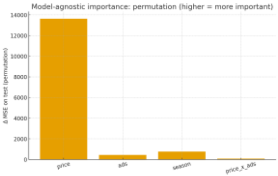
Prediction Intervals (concept)

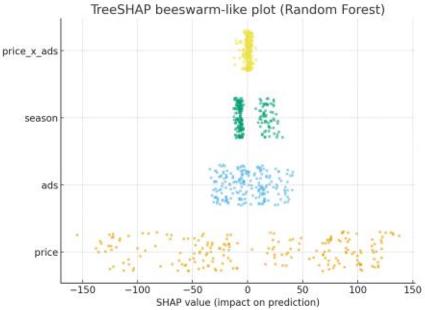
- Communicate uncertainty around point forecasts.
- Use residual distribution and model variance.

Feature Importance — Options

- Coefficients (standardized) for linear models.
- Permutation importance: ΔMSE on shuffled feature.
- Partial dependence/ICE for non-linear models (concept).







SHAP — Concepts

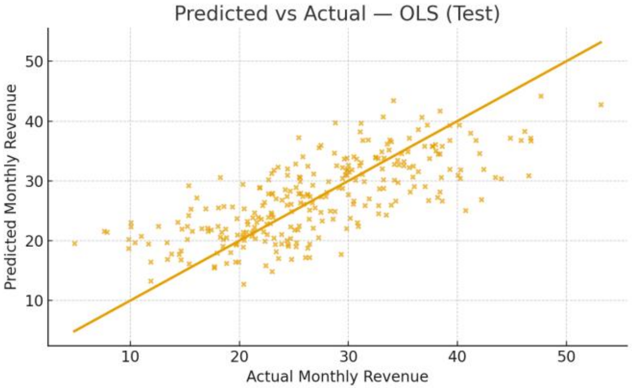
- Shapley values attribute prediction to features.
- Additivity & local accuracy; consistency across models.
- For linear models with mean reference: contribution ≈ β·(x−E[x]).

Case Setup — Questions

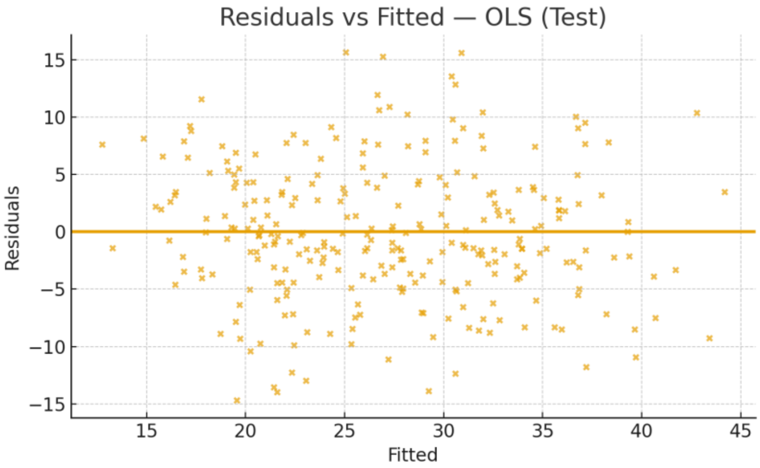
- Which levers (usage, plan) drive revenue most?
- What’s the expected lift from converting to Postpaid?
- How do complaints impact revenue holding others fixed?

Predicted vs Actual — OLS (Test)

Good models concentrate around the diagonal

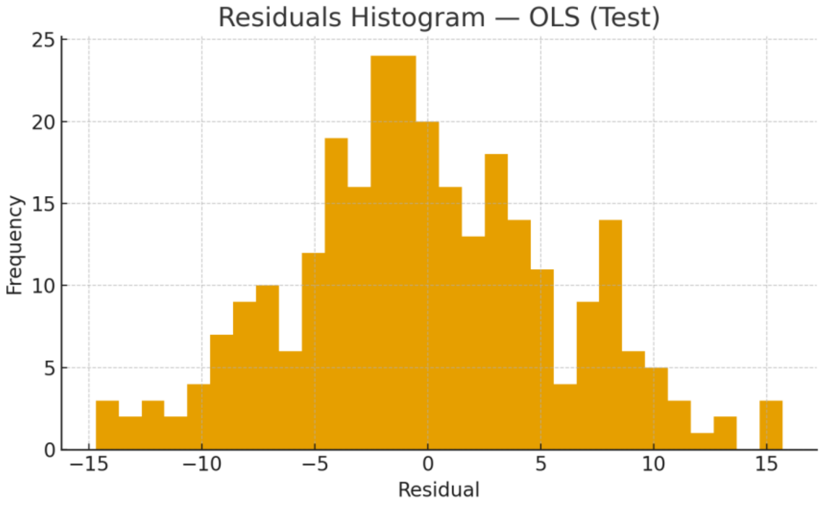


Residuals vs Fitted — OLS



Look for patterns → misspecification or heteroscedasticity

Residuals Histogram — OLS



Transformations & Binning (when to)

- Heavy-tailed predictors → log/Box-Cox/Yeo-Johnson.
- Capped bins for extreme outliers; domain caps.

Interactions & Nonlinearity (business logic first)

- E.g., Postpaid × Usage interaction for pricing effects.
- Avoid fishing — pre-specify hypotheses where possible.

Robust Regression (concept)

- Huber/quantile regression for heavy tails or outliers.
- Trade off robustness vs efficiency.

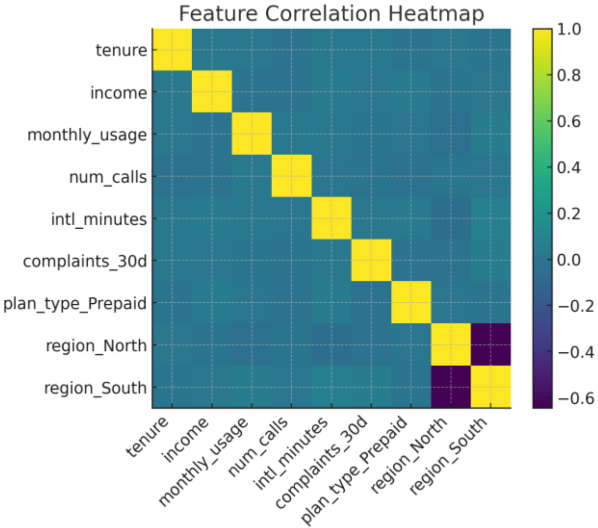
Categorical Encoding for Regression

- Dummy coding (drop one) vs effects coding.
- Beware of the dummy variable trap.

Model Diagnostics — Checklist

- Spec, residuals, influence, collinearity, stability over time.
- Retrain and revalidate periodically (data drift).

Feature Correlation Heatmap



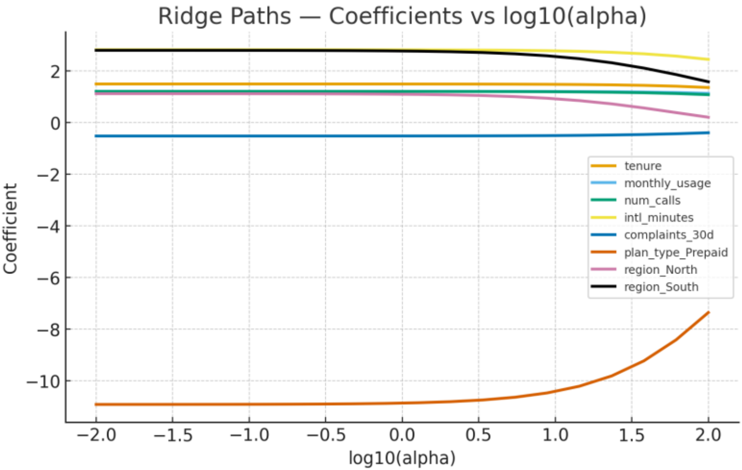
Identify highly correlated predictors

Ridge Paths — Reading the Plot

- Coefficients shrink smoothly as λ increases.
- Helps stabilize estimates under multicollinearity.

Ridge Coefficient Paths

Top 8 coefficients vs log10(alpha)

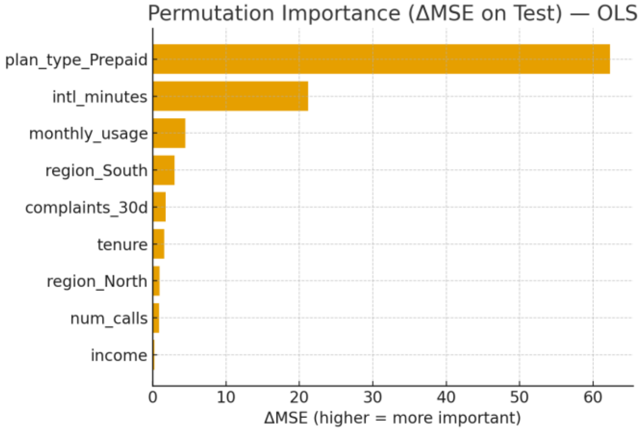


Permutation Importance — Interpretation

- ΔMSE after shuffling a feature approximates its importance.
- Model-agnostic; caution with correlated features.

Permutation Importance — OLS (Test)

Top 10 features by ΔMSE

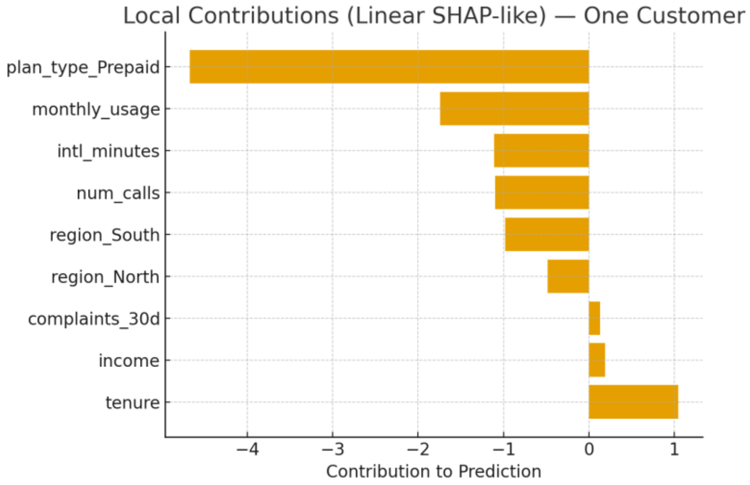


Local Explanation — Example

- Decompose one prediction into feature contributions.
- Baseline: mean prediction; contributions sum to deviation.

Local Contributions (Linear SHAP-like)

$\beta \cdot (x - E[x])$ for standardized features



Code — Closed-form OLS (NumPy)

```
import numpy as np
# X has intercept column appended as last column
beta_hat = np.linalg.pinv(X_train.T @ X_train) @ (X_train.T @
y_train)
yhat = X_test @ beta_hat
rmse = np.sqrt(np.mean((y_test - yhat)**2))
```

Code — Ridge Path (Closed-form)

```
alphas = np.logspace(-2, 2, 20)
I = np.eye(Xc_train.shape[1])
coefs = []
for a in alphas:
    beta_r = np.linalg.pinv(Xc_train.T @ Xc_train + a*I) @
(Xc_train.T @ y_center_train)
    coefs.append(beta_r)
```

Code — Permutation Importance (ΔMSE)

```
def perm_importance(model_beta, X_test, y_test, features):
    base = np.mean((y_test - X_test @ model_beta)**2)
    out = []
    for col in features:
        Xp = X_test.copy()
        j = features.index(col)
        Xp[:, j] = np.random.permutation(Xp[:, j])
        delta = np.mean((y_test - Xp @ model_beta)**2) - base
        out.append((col, delta))
    return sorted(out, key=lambda x: x[1], reverse=True)
```

Case Study — Revenue Model (Results)

- Test RMSE (OLS): 5.90
- Top drivers by standardized coefficients & permutation importance.
- Business insight: quantify lift from plan conversion and usage.

Hands-on Tasks

- Recreate OLS; add an interaction and compare RMSE.
- Tune Ridge/Lasso (grid of λ); pick best via CV.
- Produce a one-page model card (data, metrics, caveats).

Quick Quiz (10)

- Name two OLS assumptions and how to check them.
- When choose Ridge vs Lasso?
- Why standardize features before regularization?
- How to interpret permutation importance with correlated features?
- What does a SHAP bar convey for one prediction?

Key Takeaways

- Start from business levers; keep models as simple as possible.
- Regularization improves stability and generalization.
- Explainability (coeffs, permutation, SHAP) builds trust in decisions.

Recommended References

- James et al. — ISLR.
- Hastie/Tibshirani/Wainwright — Statistical Learning with Sparsity.
- Kuhn & Johnson — Applied Predictive Modeling.
- NumPy & Matplotlib documentation.

Appendix — Matrix Notes

- Hat matrix $H = X(X^T X)^{-1} X^T$; residuals $e = (I - H)y$.
- Ridge solution: $\beta = (X^T X + \lambda I)^{-1} X^T y$; no penalty on intercept.

Appendix — Lasso Geometry (concept)

- L1 ball corners → sparse solutions.
- Group correlation can cause coefficient instability.