Statistics and Data Analysis
Unit 05 – Lecture 01 Notes
Feature Selection, Engineering and Dimensionality Reduction (Intro)

Tofik Ali

February 17, 2026

Topic

Intro to feature selection, feature engineering, and dimensionality reduction.

How to Use These Notes

These notes are written for students who are seeing the topic for the first time. They follow the slide order, but add the missing 'why', interpretation, and common mistakes. If you get stuck, look at the worked exercises and then run the Python demo.

Course repository (slides, demos, datasets): `https://github.com/tali7c/Statistics-and-Data-Analysis`

Time Plan (55 minutes)

- 0–10 min: Attendance + recap of previous lecture
- 10–35 min: Core concepts (this lecture's sections)
- 35–45 min: Exercises (solve 1–2 in class, rest as practice)
- 45–50 min: Mini demo + interpretation of output
- 50–55 min: Buffer / wrap-up (leave 5 minutes early)

Slide-by-slide Notes

Title Slide

State the lecture title clearly and connect it to what students already know. Tell students what they will be able to do by the end (not just what you will cover).

Quick Links / Agenda

Explain the structure of the lecture and where the exercises and demo appear.

- Overview
- Why Features
- Selection vs Reduction
- Exercises
- Demo
- Summary

Learning Outcomes

- Differentiate feature selection vs dimensionality reduction
- Explain why too many features can hurt (overfitting, cost)
- Describe a simple feature engineering pipeline
- Identify target leakage in engineered features

Why these outcomes matter. **Data leakage** happens when information from the future or from the test set influences training. Typical examples: scaling before splitting, using target-related features, or using random splits for time series. Leakage can produce very good-looking accuracy that disappears in real deployment.

Why Features: Key Points

- Features are how models see data
- Goal: represent signal and reduce noise
- Bad features -> bad models

Selection vs Reduction: Key Points

- Selection keeps a subset of original features
- Reduction creates new components (e.g., PCA)
- Validate choices using CV

Explanation. **PCA** finds new axes (principal components) that capture maximum variance. It is a rotation of the feature space. Because PCA is variance-based, it is sensitive to scaling: standardize features first unless all features are already comparable.

Exercises (with Solutions)

Attempt the exercise first, then compare with the solution. Focus on interpretation, not only arithmetic.

Exercise 1: Selection or reduction

Dropping 30 out of 100 features is selection or reduction?

Solution

- Feature selection (subset).

Exercise 2: Leakage

Is using final exam score to predict final grade leakage?

Solution

- Yes; it contains future/target information.

Walkthrough. Data leakage happens when information from the future or from the test set influences training. Typical examples: scaling before splitting, using target-related features, or using random splits for time series. Leakage can produce very good-looking accuracy that disappears in real deployment.

Exercise 3: Engineering example

Give one time-based engineered feature.

Solution

- Day-of-week, month, time-since-last-event, rolling average, etc.

Mini Demo (Python)

Run from the lecture folder:

```
python demo/demo.py
```

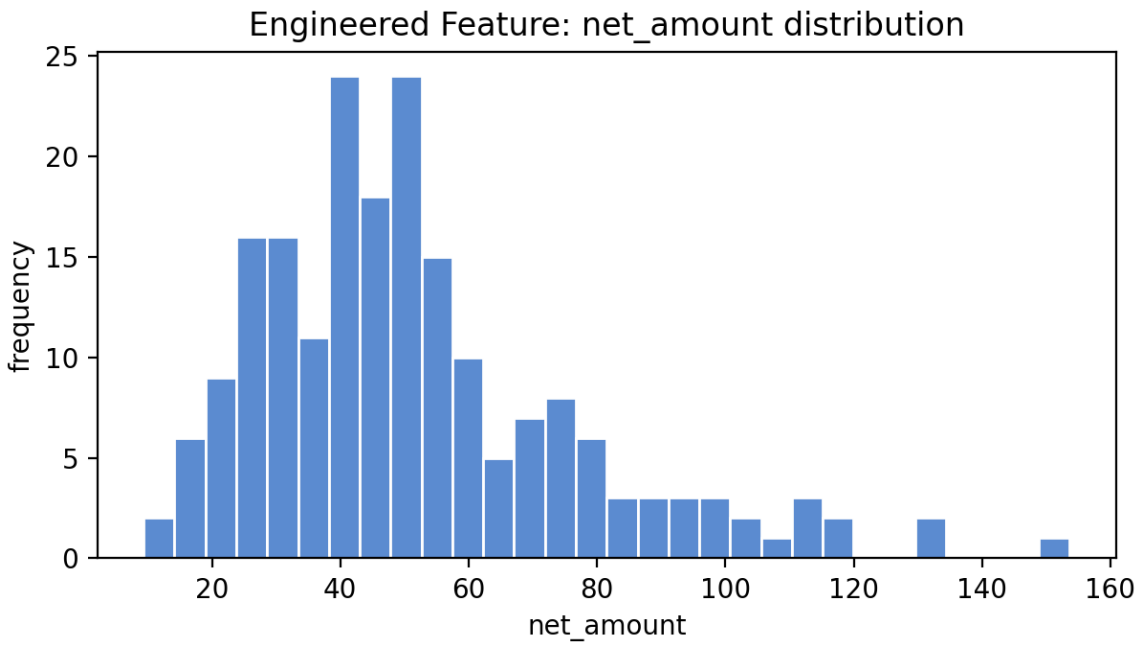
Output files:

- `images/demo.png`
- `data/results.txt`

What to show and say.

- Shows how feature quality affects model performance on a toy dataset.
- Creates simple engineered features and compares baseline vs improved model.
- Use it to motivate selection/reduction to fight overfitting and cost.

Demo Output (Example)



Summary

- Key definitions and the main formula.
- How to interpret results in context.
- How the demo connects to the theory.

Exit Question

Why can adding more features sometimes reduce test accuracy?

Suggested answer (for revision). Extra features can add noise and increase overfitting (curse of dimensionality), reducing test performance even if training fit improves.

References

- Montgomery, D. C., & Runger, G. C. *Applied Statistics and Probability for Engineers*, Wiley.
- Devore, J. L. *Probability and Statistics for Engineering and the Sciences*, Cengage.
- McKinney, W. *Python for Data Analysis*, O'Reilly.

Appendix: Slide Deck Content (Reference)

The material below is a reference copy of the slide deck content. Exercise solutions are explained in the main notes where applicable.

Title Slide

https://github.com/tali7c/Statistics-and-Data-Analysis

Quick Links

Overview Why Features Selection vs Reduction Exercises Demo Summary

Agenda

- Overview
- Why Features
- Selection vs Reduction
- Exercises
- Demo
- Summary

Learning Outcomes

- Differentiate feature selection vs dimensionality reduction
- Explain why too many features can hurt (overfitting, cost)
- Describe a simple feature engineering pipeline
- Identify target leakage in engineered features

Why Features: Key Points

- Features are how models see data
- Goal: represent signal and reduce noise
- Bad features -> bad models

Selection vs Reduction: Key Points

- Selection keeps a subset of original features
- Reduction creates new components (e.g., PCA)
- Validate choices using CV

Exercise 1: Selection or reduction

Dropping 30 out of 100 features is selection or reduction?

Solution 1

- Feature selection (subset).

Exercise 2: Leakage

Is using final exam score to predict final grade leakage?

Solution 2

- Yes; it contains future/target information.

Exercise 3: Engineering example

Give one time-based engineered feature.

Solution 3

- Day-of-week, month, time-since-last-event, rolling average, etc.

Mini Demo (Python)

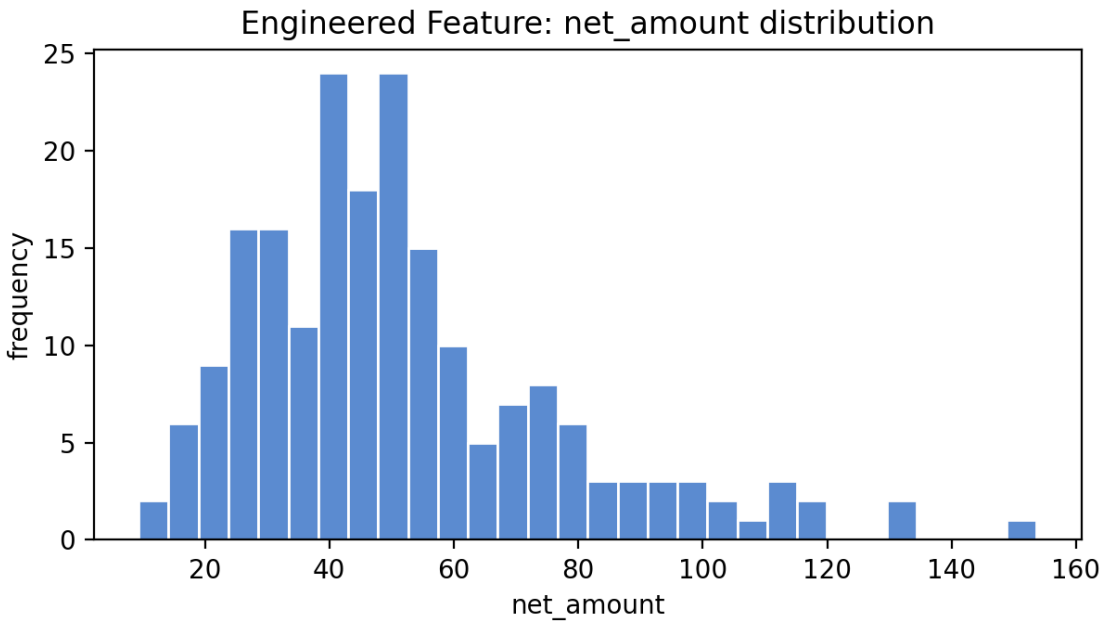
Run from the lecture folder:

```
python demo/demo.py
```

Outputs:

- `images/demo.png`
- `data/results.txt`

Demo Output (Example)



Summary

- Key definitions and the main formula.
- How to interpret results in context.
- How the demo connects to the theory.

Exit Question

Why can adding more features sometimes reduce test accuracy?