Course Project Guidelines for SF2935 (Autumn 2025)

You are welcome to choose a topic in any area of machine learning or statistics related to the course. You are strongly encouraged to choose a topic that you would like to learn more about, rather than one you are already familiar with. It is fine to choose a topic related to your independent research, though you must choose an aspect for the course project that you would not have done otherwise.

The main goal is to move beyond textbook exercises and engage with the complexities of real-world data. A successful project will weave together theoretical concepts, algorithmic implementation, and empirical analysis into a coherent narrative.

1 Core Requirement: Demonstration of Statistical Principles

A central requirement for the project is the demonstration and application of statistical principles, either those learned in this course or new ones you explore.

- For **theoretical or algorithmic projects**, you must analyze an algorithm's properties, discussing the theoretical assumptions under which it should work and the conditions under which it will fail, or prove results related to its statistical behavior.
- For **experimental projects**, you must provide insight as to why certain algorithms perform better than others by connecting their performance to underlying statistical theory (e.g., bias-variance trade-off, effects of regularization, curse of dimensionality). Simply applying many algorithms to one dataset and reporting test results does not constitute a good project.

2 Project Components

The project can be completed individually or in group (highly recommended) of up to five students.

2.1 Project Proposal

A 1–2 page proposal is due on **September 12**. It should contain:

- Project title and a 1–2 sentence summary.
- Motivation: Why is this problem important or interesting?
- A precise description of the question you are trying to answer and your proposed methodology.
- Data: A clear description of the dataset(s) you will use.
- A preliminary reading list of key papers.

2.2 Final Report

A final report is due on **October 17**. The report should be 8–10 pages (excluding references and appendix) and formatted using the NeurIPS 2025 LaTeX template[1]. It should be scientifically sound, with a clear structure including:

- **Introduction:** Present the problem, motivation, and key research questions.

[1] Available at https://neurips.cc/Conferences/2025/CallForPapers.

- **Methodology:** Describe your models, algorithms, or experimental design in detail.
- **Results:** Present empirical findings, theoretical results (if any) and their empirical validations, visualizations, and quantitative analyses.
- **Conclusion:** Summarize insights, limitations, and possible future directions.
- **Appendix (optional but encouraged):** Include additional derivations, proofs, extended experiments, or detailed explanations that support the main text without disrupting its flow.

2.3 Code Repository

The final report should also provide a link to a well-documented code repository (e.g., GitHub) containing your implementation, experiments, and instructions to reproduce your results.

3 Project Suggestions

Here are several project ideas designed to be both challenging and rewarding. They span different domains and are framed to encourage engagement with core statistical principles. You are welcome to propose new project ideas, as long as they are clearly motivated and grounded in statistical principles.

3.1 Study and Prepare Lecture Notes on a Topic Not Covered in Class

Your task is to go in-depth on a topic we did not have time to cover, mastering the theory and presenting it clearly. Your report will serve as a set of lecture notes on the topic.

1. **Gaussian Processes for Regression and Classification**
   - *Statistical Principles*: Bayesian inference, kernel methods, stochastic processes, defining priors over functions, marginal likelihood for hyperparameter tuning.
   - *Possible Project*: Introduce Gaussian Processes (GP) and study their statistical properties. Implement GP classification and regression models from scratch, and apply them on real-world datasets[2]. Demonstrate how the choice of the kernel (e.g., RBF vs. Matérn) acts as a prior that controls the smoothness of the resulting function.
2. **Self-Supervised and Contrastive Learning**
   - *Statistical Principles*: Mutual information estimation, inductive bias, generalization, role of augmentations as implicit priors.
   - *Possible Project*: Introduce self-supervised learning from a statistical learning theory perspective. Include the role of data augmentations, the connection to mutual information, and contrastive vs. predictive paradigms. Demonstrate on a real-world dataset (e.g., Fashion-MNIST[3]) how a learned representation can improve downstream supervised tasks.

3.2 Address a Problem from a Competition or Large-Scale Dataset

The goal here is a deep, comparative analysis on a challenging, real-world dataset, with a focus on connecting performance to statistical theory. There are many places where you can find publicly available real-world datasets, for instance `https://www.kaggle.com/datasets`.

[2] `https://gaussianprocess.org/gpml/data/`

[3] `https://github.com/zalandoresearch/fashion-mnist`

1. **Computer Vision: Neural Networks vs. Kernel Methods**
   - *Statistical Principles*: Deep neural networks, kernel methods in high dimensions, bias-variance trade-off, regularization, the role of priors.
   - *Dataset Suggestion*: CheXpert[4] dataset or an interesting one from Kaggle[5].

2. **Natural Language Processing: Unsupervised vs. Supervised Text Classification**
   - *Statistical Principles*: Generative vs. discriminative modeling, bag-of-words assumption, high-dimensional geometry and the margin.
   - *Dataset Suggestion*: 20 Newsgroups[6] dataset or the arXiv abstract[7] dataset.

3.3 Analyze an Existing or New Algorithm

For those with a strong background in optimization or theory. In addition to theoretical analysis, the studied algorithm should be implemented and tested on real-world datasets.

1. **Proximal Gradient Methods for Structured Sparsity**
   - *Statistical Principles*: Regularization, sparsity, convex optimization, structured priors.
   - *Possible Project*: Reproduce the results in the original paper[8] that proposed the Fused Lasso and apply the algorithm to a signal denoising problem.

2. **Analysis of SGD with a Large Initial Learning Rate for Training Neural Networks**
   - *Statistical Principles*: Implicit regularization, generalization properties of algorithms, non-convex optimization.
   - *Possible Project*: Reproduce the key results in the original paper[9]. Train a neural net with large initial learning rate and annealing, and show that this achieves improved generalization compared to the same network trained with a small learning rate from the start.

3.4 Acquire an Interesting Dataset and Learn Something Insightful

This is an open-ended option for creative data-driven exploration, with a strong emphasis on connecting the analysis to statistical principles.

1. **Optiver Realized Volatility Prediction**
   The dataset by Optiver[10] comprises high-frequency stock data (book and trade records) for over 100 stocks. The task is to predict the short-term realized volatility for each stock within a fixed time window (e.g., 10 minutes), using past observations.
   - *Statistical Principles*: Stationarity vs. non-stationarity, bias–variance trade-off under noisy data, feature engineering for volatility, evaluation using RMSPE.
   - *Possible Project*: Engineer features capturing volatility clustering and market microstructure effects. Compare classical statistical models (GARCH, ARIMA) against machine learning approaches (e.g., gradient boosting, neural nets, Transformers) for volatility forecasting. Analyze why some models generalize better or are more robust under changing market conditions.

[4] https://stanfordmlgroup.github.io/competitions/chexpert/

[5] https://www.kaggle.com/datasets?tags=13207-Computer+Vision

[6] https://archive.ics.uci.edu/dataset/113/twenty+newsgroups

[7] https://www.kaggle.com/datasets/spsayakpaul/arxiv-paper-abstracts

[8] https://stanford.edu/group/SOL/papers/fused-lasso-JRSSB.pdf

[9] https://arxiv.org/abs/1907.04595

[10] https://www.kaggle.com/competitions/optiver-realized-volatility-prediction/overview

2. **Abstraction and Reasoning Challenge (ARC)**
   The ARC dataset[11] consists of 800 tasks designed to test an agent's ability to recognize patterns and solve novel problems with very limited examples. Each task contains a small set of input–output pairs that define a transformation rule or pattern, and the goal is to predict outputs for new inputs following the same rule. Tasks vary widely in difficulty and often require reasoning over shapes, colors, numbers, and spatial arrangements.
   - *Statistical Principles*: Generalization, few-shot learning, inductive bias.
   - *Possible Project*: Implement or evaluate models that attempt to solve ARC tasks unseen during training, focusing on generalization rather than memorization. Propose statistically principled metrics to assess generalization across unseen tasks. Analyze how different model architectures and inductive biases affect their ability to generalize from a few examples to novel inputs.

[11] `https://www.kaggle.com/c/abstraction-and-reasoning-challenge`