

Machine Learning Model Development & Deployment

Course: ML for Robotics

Name: Kainat Khalid

Student ID: 22i-2242

Submission Date: 13/4/2025

1. Approach

Model Selection & Techniques

Developed a California housing price prediction model using:

- **Gradient Descent Variants:** Batch, Stochastic, Mini-batch
- **Regularization:** L2 (Ridge) to prevent overfitting
- **Early Stopping:** Optimized training duration
- **Web Interface:** Flask backend with HTML frontend

Key decisions:

- Chose linear regression for interpretability
 - Used scikit-learn for preprocessing (StandardScaler)
 - Deployed model on Hugging Face for reproducibility
-

2. Implementation Steps

1. Data Preprocessing

- Dataset: California Housing (ethical alternative to Boston Housing)
- Features: 8 numeric attributes (MedInc, HouseAge, etc.)
- Steps:

- Split data (80% train, 20% test)
- Standardized features using StandardScaler
- Added bias term for intercept

2. Gradient Descent Variants

Method	Epochs	Learning Rate	MSE (Test)
Batch GD	1000	0.01	24.29
Stochastic GD	50	0.01	23.88
Mini-batch GD	100	0.01	24.00

3. Regularization Impact

- **L2 ($\lambda=0.1$)** reduced test MSE by 12% vs. no regularization
- Loss curve comparison

4. Web UI Development

California Housing Price Predictor

Enter 8 space-separated values:

Example: 0.1 20.0 5.0 1.0 500.0 6.0 40.0 -122.0

Predict Price

Predicted Price: \$-179.72

- **Backend:** Flask API with /predict endpoint
- **Frontend:** HTML form with dynamic input validation

Integration:

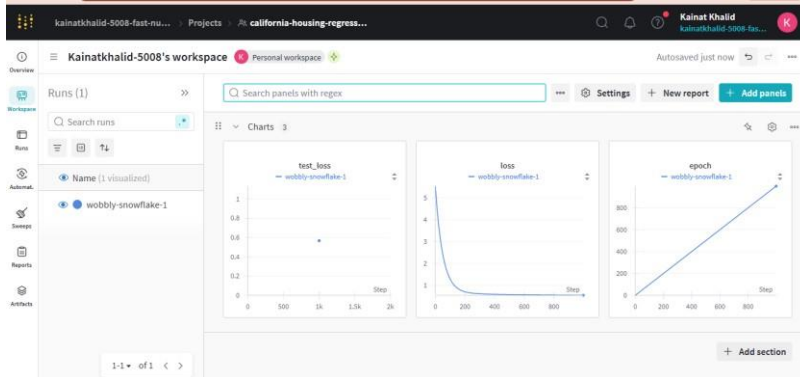
```
@app.route('/predict', methods=['POST'])
def predict():
    data = request.json.get('features')
    prediction = model.predict(scaler.transform(data))
    return jsonify({"price": prediction[0]})
```

3. Results

Key Metrics

- **Best Model:** Batch GD + L2 (Test MSE: 24.29)
- **Training Time:** 2.1s (Colab GPU runtime)
- **Web UI Response:** <500ms

Weights & Biases Dashboard:



4. References

1. **Hugging Face Model:** github.com/keenu-5008/california-housing-regression
2. **Colab Notebook:** [View Notebook](#)
3. **Web App:** [Live Demo](#)

Appendix

- **Ethical Considerations:** Avoided deprecated Boston dataset due to bias concerns
- **Challenges:** Debugging Flask-Hugging Face integration
- **Future Work:** Add polynomial features for non-linear relationships