Andrew Ng's Lecture on Applying Machine Learning

Core Diagnostic Framework: Bias vs. Variance

- **High Variance (Overfitting)**: Training error $\ll$ test error
  - Fixes: More training data, fewer features, increase λ
- **High Bias (Underfitting)**: Both training and test error high
  - Fixes: More features, polynomial features, decrease λ
- **Diagnostic**: Plot learning curves (error vs. training set size m)

Additional Diagnostics

- **Optimization vs. Objective Function**:
  - If $J(\theta_{SVM}) > J(\theta_{LR})$, $a(\theta_{SVM}) > a(\theta_{LR})$ $\rightarrow$ optimization problem
  - If $J(\theta_{SVM}) \leq J(\theta_{LR})$, $a(\theta_{SVM}) > a(\theta_{LR})$ $\rightarrow$ wrong objective
- **Error Analysis**: Replace pipeline components with ground truth, measure impact
- **Ablative Analysis**: Remove features/components, measure degradation

Getting Started: 2 Approaches

1. **Careful Design**: Perfect features upfront $\rightarrow$ Risk: premature optimization
2. **Build-and-Fix** (Recommended): Quick implementation $\rightarrow$ diagnostics $\rightarrow$ targeted fixes

Practical Workflow

1. Implement baseline quickly
2. Run diagnostics to identify problem type
3. Apply appropriate fixes:
   - Variance problem $\rightarrow$ regularization/data/features
   - Bias problem $\rightarrow$ complexity/features
   - Optimization problem $\rightarrow$ algorithm/iterations
4. Iterate based on diagnostic results

Key Principles

- **Diagnostics > guessing**: Time spent on diagnostics is time well spent
- **Start simple**: Plot data first, implement baseline, then improve
- **Avoid premature optimization**: You can't know what needs work until you try
- **Let data guide you**: Error analysis reveals where to focus effort

A Few Useful Things to Know About Machine learning

Core Ideas

- Learning = Representation + Evaluation + Optimization
- Goal: Generalize to new data, not memorize training data
- Data alone isn't enough – algorithms need built-in assumptions
- Overfitting is the main enemy

Practical Tips

- More data > fancier algorithms
- Feature engineering matters most
- Combine multiple models for better results
- Start with simple algorithms first

Common Mistakes

- High dimensions break intuition
- Theoretical guarantees are too pessimistic
- Simple ≠ accurate
- Can represent ≠ can learn

What Works

- Use diagnostics to fix problems
- Cross-validate carefully
- Remember: correlation ≠ causation
- Optimize for human time, not just accuracy