Kaggle Competition Project Structure

Competition-optimized structure with experiment tracking, ensembles, and submission workflow.

#ml #python #kaggle #competition #data-science

PNG PDF Copy Prompt

Project Directory

```
competition/
  notebooks/ Exploration and…
    eda.ipynb Exploratory dat…
    baseline.ipynb First submission
    experiments/
      exp_001_xgb.ipynb
  src/ Reusable code
    __init__.py
    data.py Data loading an…
    features.py Feature enginee…
    models.py Model definitio…
    train.py Training loop
    inference.py Test predictions
    ensemble.py Ensemble methods
  input/ Competition dat…
    train.csv
    test.csv
    sample_submission.csv
  output/ Predictions and…
    models/
    oof/ Out-of-fold pre…
    submissions/
  configs/ Experiment conf…
    exp_001.yaml
  requirements.txt
  .gitignore
  README.md Competition not…
```

Why This Structure?

Optimized for competition workflow: fast iteration, OOF predictions for stacking, and organized submissions. The `input/` folder mirrors Kaggle's structure. Experiments are numbered (`exp_001`) for easy tracking. `ensemble.py` combines model predictions.

Key Directories

`notebooks/experiments/` - Numbered experiment notebooks

`src/ensemble.py` - Blending, stacking, averaging

`output/oof/` - Out-of-fold predictions for stacking

`output/submissions/` - Dated submission files

`configs/` - YAML configs per experiment

Experiment Naming

```
# Naming convention
exp_001_baseline_xgb.ipynb
exp_002_lgbm_tuned.ipynb
exp_003_nn_tabular.ipynb
exp_004_ensemble_blend.ipynb

# Submissions
sub_exp001_0.812_20241215.csv
```

When To Use This

- Kaggle and similar ML competitions
- Hackathons with submission deadlines
- Ensemble and stacking workflows
- Rapid experimentation cycles

Trade-offs

`Not production-ready` - Optimized for score, not deployment

`Messy history` - Fast iteration over clean commits

`Local focus` - May need GPU setup on Kaggle