
H2O.ai

GLM Best Practices

- Regularization Selection
  - Explore a few values for alpha, e.g. 0.01, 0.25, 0.5, 0.75, 0.99
- Wide Data Sets (10K+ columns)
  - Iteratively Reweighted Least Squares (IRLS) fails with lambda = 0
    - IRLS requires p x p Hessian matrix, where p = # of coefficients
    - Could use Limited-memory BFGS (L-BFGS)
  - IRLS + lambda search works and is recommended
    - Use alpha >> 0
    - Can produce 1K+ non-zero coefficients
  - L-BFGS + L2 penalty works
  - L-BFGS + L1 penalty works, but may take a long time

GLM Example

scikit-learn-like interface for modeling

```
y = "income"
x = ["age", "workclass", "fnlwgt", "education", "marital-status", "occupation", "relationship",
     "race", "sex", "capital-gain", "capital-loss","hours-per-week", "native-country"]
```

```
from h2o.estimators.glm import H2OGeneralizedLinearEstimator
glm_0 = H2OGeneralizedLinearEstimator(family = "binomial", lambda_search = True,
                                      nfolds = 5, seed = 123)
glm_0.train(x = x, y = y, training_frame = census_data, model_id = "income_glm_0")
```

```
glm Model Build progress: |████████████████████████████████████████████████| 100%
```

```
from h2o.grid.grid_search import H2OGridSearch
glm_hyper_parameters = {"alpha": [0.5, 0.75, 1]}
glm_grid = H2OGridSearch(H2OGeneralizedLinearEstimator(family = "binomial", lambda_search = True,
                                                       nfolds = 5, seed = 123),
                         glm_hyper_parameters)
glm_grid.train(x = x, y = y, training_frame = census_data, grid_id = "income_glm_grid")
```

```
glm Grid Build progress: |████████████████████████████████████████████████| 100%
```

GLM Best Practices

- Regularization Selection
  - Explore a few values for alpha, e.g. 0.01, 0.25, 0.5, 0.75, 0.99
- Wide Data Sets (10K+ columns)
  - Iteratively Reweighted Least Squares (IRLS) fails with lambda = 0
    - IRLS requires p x p Hessian matrix, where p = # of coefficients
    - Could use Limited-memory BFGS (L-BFGS)
  - IRLS + lambda search works and is recommended
    - Use alpha >> 0
    - Can produce 1K+ non-zero coefficients
  - L-BFGS + L2 penalty works
  - L-BFGS + L1 penalty works, but may take a long time