

Benemérita Universidad Autónoma de Puebla
Facultad de Ciencias de la Computación

Máquinas de aprendizaje

Reporte: Mejora del rendimiento del modelo, parte II



BUAP

Docente: Abraham Sánchez López

Alumno

Taisen Romero Bañuelos

Matrícula

202055209

Mejora del rendimiento del modelo

El documento incluye varios enfoques y técnicas con el fin de mejorar los resultados obtenidos. Como es de costumbre, hay métricas de precisión de las que no nos podemos confiar, por lo que Kappa será nuestra medida más confiable a considerar. En este sentido, según avanzamos vimos métodos que poco a poco mejoraba los resultados, y tiene sentido que conforme sean más sofisticados los modelos mejor sean sus resultados. Por ejemplo, XGB incluye la metodología de partir el conjunto de entrenamiento original en otros dos conjuntos para entrenar unos modelos “temporales”. De hecho, se me ocurrió que para mejorar el Kappa de XGB podría usar apilamiento con los modelos propios del PDF (ya que eran muy eficientes todos), sin embargo, no logré hacer que funcionara el experimento. En su lugar, hice un grid search (rejilla) para combinar los hiperparámetros y **encontrar la combinación ideal**. Ya lleva un rato ejecutando la combinatoria, así que le daré un tiempo de 30 minutos al grid para ver si logra encontrar la combinación ideal.

De mientras, hablemos de la interpretación de los resultados usando XGB como un factor común para la comparación. Definitivamente XGB tiene el Kappa más elevado (0.423), pero, ¿qué hay de GBM (su primo cercano)?, pues, tiene una buena capacidad, pero quizá su configuración por defecto de generar 100 árboles esté limitando un poco su potencial (aunque se menciona que las mejoras no son tan significativas).

En cuanto a Random Forest, suele tener un rendimiento estable en muchos contextos, en este caso particular parece que no logra capturar con la misma precisión los patrones subyacentes del dataset. Esto podría deberse a que su enfoque basado en árboles independientes pierde cierta sensibilidad ante relaciones sutiles entre variables. Esto es algo que los métodos de boosting, como XGB, parecen explotar mejor al corregir errores de forma iterativa.

Bueno, luego de casi 45 minutos por fin terminó el grid.

```
> print(m_xgb_fine)
extreme Gradient Boosting
```

```
1000 samples
 16 predictor
 2 classes: 'no', 'yes'
```

```
No pre-processing
Resampling: Cross-validated (10 fold)
Summary of sample sizes: 900, 900, 900, 900, 900, 900, ...
Resampling results across tuning parameters:
```

eta	max_depth	gamma	colsample_bytree	min_child_weight	subsample	nrounds	Accuracy	Kappa
0.05	2	0.0	0.6	1	0.75	100	0.731	0.2423301
0.05	2	0.0	0.6	1	0.75	150	0.739	0.2868815
0.05	2	0.0	0.6	1	0.75	200	0.741	0.3065040
0.05	2	0.0	0.6	1	0.85	100	0.719	0.2105830
0.05	2	0.0	0.6	1	0.85	150	0.732	0.2693462
0.05	2	0.0	0.6	1	0.85	200	0.741	0.3059071
0.05	2	0.0	0.6	1	0.90	100	0.728	0.2333874
0.05	2	0.0	0.6	1	0.90	150	0.743	0.2939400
0.05	2	0.0	0.6	1	0.90	200	0.739	0.3000080
0.05	2	0.0	0.6	1	1.00	100	0.729	0.2180773
0.05	2	0.0	0.6	1	1.00	150	0.731	0.2519782
0.05	2	0.0	0.6	1	1.00	200	0.741	0.3028723
0.05	2	0.0	0.6	5	0.75	100	0.726	0.2356354
0.05	2	0.0	0.6	5	0.75	150	0.737	0.2859806
0.05	2	0.0	0.6	5	0.75	200	0.752	0.3406069
0.05	2	0.0	0.6	5	0.85	100	0.720	0.2046023
0.05	2	0.0	0.6	5	0.85	150	0.732	0.2675148
0.05	2	0.0	0.6	5	0.85	200	0.744	0.3189256
0.05	2	0.0	0.6	5	0.90	100	0.729	0.2308528
0.05	2	0.0	0.6	5	0.90	150	0.731	0.2642310
0.05	2	0.0	0.6	5	0.90	200	0.742	0.3130589
0.05	2	0.0	0.6	5	1.00	100	0.719	0.1920963
0.05	2	0.0	0.6	5	1.00	150	0.725	0.2414952
0.05	2	0.0	0.6	5	1.00	200	0.744	0.3155231
0.05	2	0.0	0.6	10	0.75	100	0.727	0.2331102
0.05	2	0.0	0.6	10	0.75	150	0.736	0.2907167
0.05	2	0.0	0.6	10	0.75	200	0.738	0.3014876
0.05	2	0.0	0.6	10	0.85	100	0.729	0.2429382
0.05	2	0.0	0.6	10	0.85	150	0.734	0.2773613
0.05	2	0.0	0.6	10	0.85	200	0.736	0.2993777
0.05	2	0.0	0.6	10	0.90	100	0.720	0.2085094
0.05	2	0.0	0.6	10	0.90	150	0.743	0.3058066
0.05	2	0.0	0.6	10	0.90	200	0.739	0.3034118
0.05	2	0.0	0.6	10	1.00	100	0.729	0.2262051
0.05	2	0.0	0.6	10	1.00	150	0.740	0.2908158
0.05	2	0.0	0.6	10	1.00	200	0.741	0.3103851
0.05	2	0.0	0.7	1	0.75	100	0.729	0.2428967
0.05	2	0.0	0.7	1	0.75	150	0.746	0.3076193
0.05	2	0.0	0.7	1	0.75	200	0.747	0.3264177
0.05	2	0.0	0.7	1	0.85	100	0.720	0.2108199
0.05	2	0.0	0.7	1	0.85	150	0.732	0.2727450
0.05	2	0.0	0.7	1	0.85	200	0.739	0.3067413
0.05	2	0.0	0.7	1	0.90	100	0.719	0.2066621
0.05	2	0.0	0.7	1	0.90	150	0.731	0.2655829
0.05	2	0.0	0.7	1	0.90	200	0.741	0.3084504
0.05	2	0.0	0.7	1	1.00	100	0.724	0.2050644
0.05	2	0.0	0.7	1	1.00	150	0.733	0.2609993
0.05	2	0.0	0.7	1	1.00	200	0.737	0.2922471
0.05	2	0.0	0.7	5	0.75	100	0.721	0.2187858
0.05	2	0.0	0.7	5	0.75	150	0.734	0.2814310
0.05	2	0.0	0.7	5	0.75	200	0.738	0.3018484
0.05	2	0.0	0.7	5	0.85	100	0.727	0.2361102
0.05	2	0.0	0.7	5	0.85	150	0.734	0.2765769
0.05	2	0.0	0.7	5	0.85	200	0.745	0.3205173
0.05	2	0.0	0.7	5	0.90	100	0.718	0.2049732
0.05	2	0.0	0.7	5	0.90	150	0.726	0.2528575
0.05	2	0.0	0.7	5	0.90	200	0.741	0.3104679
0.05	2	0.0	0.7	5	1.00	100	0.720	0.1918197

```

0.05 2      0.0  0.7      5      1.00    100    0.720    0.1918197
0.05 2      0.0  0.7      5      1.00    150    0.730    0.2554107
0.05 2      0.0  0.7      5      1.00    200    0.738    0.2982835
0.05 2      0.0  0.7     10      0.75    100    0.726    0.2336007
0.05 2      0.0  0.7     10      0.75    150    0.737    0.2898036
0.05 2      0.0  0.7     10      0.75    200    0.741    0.3100535
0.05 2      0.0  0.7     10      0.85    100    0.733    0.2534657
0.05 2      0.0  0.7     10      0.85    150    0.741    0.3002712
0.05 2      0.0  0.7     10      0.85    200    0.743    0.3176433
0.05 2      0.0  0.7     10      0.90    100    0.728    0.2409822
0.05 2      0.0  0.7     10      0.90    150    0.732    0.2799356
0.05 2      0.0  0.7     10      0.90    200    0.745    0.3240242
0.05 2      0.0  0.7     10      1.00    100    0.723    0.2160422
0.05 2      0.0  0.7     10      1.00    150    0.734    0.2784076
0.05 2      0.0  0.7     10      1.00    200    0.742    0.3132507
0.05 2      0.0  0.8      1      0.75    100    0.729    0.2420474
0.05 2      0.0  0.8      1      0.75    150    0.739    0.2933111
0.05 2      0.0  0.8      1      0.75    200    0.747    0.3255544
0.05 2      0.0  0.8      1      0.85    100    0.717    0.2011869
0.05 2      0.0  0.8      1      0.85    150    0.735    0.2817841
0.05 2      0.0  0.8      1      0.85    200    0.742    0.3128268
0.05 2      0.0  0.8      1      0.90    100    0.721    0.2115737
0.05 2      0.0  0.8      1      0.90    150    0.735    0.2791805
0.05 2      0.0  0.8      1      0.90    200    0.745    0.3152341
0.05 2      0.0  0.8      1      1.00    100    0.725    0.2104782
0.05 2      0.0  0.8      1      1.00    150    0.734    0.2630258
0.05 2      0.0  0.8      1      1.00    200    0.736    0.2893381
0.05 2      0.0  0.8      5      0.75    100    0.725    0.2254347
0.05 2      0.0  0.8      5      0.75    150    0.738    0.2927933
0.05 2      0.0  0.8      5      0.75    200    0.745    0.3242715
0.05 2      0.0  0.8      5      0.85    100    0.726    0.2294817
0.05 2      0.0  0.8      5      0.85    150    0.733    0.2723646
0.05 2      0.0  0.8      5      0.85    200    0.744    0.3178819
0.05 2      0.0  0.8      5      0.90    100    0.723    0.2147939
0.05 2      0.0  0.8      5      0.90    150    0.728    0.2682924
0.05 2      0.0  0.8      5      0.90    200    0.745    0.3249244
0.05 2      0.0  0.8      5      1.00    100    0.721    0.1993604
0.05 2      0.0  0.8      5      1.00    150    0.731    0.2600275
0.05 2      0.0  0.8      5      1.00    200    0.737    0.2946990
0.05 2      0.0  0.8     10      0.75    100    0.729    0.2412449
0.05 2      0.0  0.8     10      0.75    150    0.732    0.2767672
0.05 2      0.0  0.8     10      0.75    200    0.738    0.3052436
0.05 2      0.0  0.8     10      0.85    100    0.731    0.2454579
0.05 2      0.0  0.8     10      0.85    150    0.735    0.2896282
0.05 2      0.0  0.8     10      0.85    200    0.736    0.2982508
0.05 2      0.0  0.8     10      0.90    100    0.732    0.2535768
0.05 2      0.0  0.8     10      0.90    150    0.742    0.3050860
0.05 2      0.0  0.8     10      0.90    200    0.743    0.3197160
0.05 2      0.0  0.8     10      1.00    100    0.730    0.2256119
0.05 2      0.0  0.8     10      1.00    150    0.732    0.2747373
0.05 2      0.0  0.8     10      1.00    200    0.739    0.3072451
0.05 2      0.1  0.6      1      0.75    100    0.732    0.2499641
0.05 2      0.1  0.6      1      0.75    150    0.742    0.2972266
0.05 2      0.1  0.6      1      0.75    200    0.744    0.3174533
[ reached getOption("max.print") -- omitted 2481 rows ]

```

kappa was used to select the optimal model using the largest value.
The final values used for the model were nrounds = 200, max_depth = 5, eta = 0.1, gamma = 0, colsample_bytree = 0.7, min_child_weight = 5 and subsample = 0.9.

Como vemos, el tiempo de espera valió un poco la pena. Aumentamos otro poco el Kappa obtenido. Pasamos de 0.423 a **0.431**.

```

> m_xgb_fine$bestTune
      nrounds max_depth eta gamma colsample_bytree min_child_weight subsample
1569      200        5 0.1    0              0.7              5         0.9
> max(m_xgb_fine$results["Kappa"])
[1] 0.4313002

```

Definitivamente el tener varios parámetros a configurar aumenta el tiempo de espera si hacemos un grid, pero viendo que obtuvimos el mejor kappa considero que vale la pena esperar, o considerar el apilamiento.