競馬人工知能を1年間育てた話

AlphaImpact
大元 司

db analytics showcase
Sapporo 2017

自己紹介 

- 大元 司 (Tsukasa OMOTO)
- https://twitter.com/henry0312
- https://github.com/henry0312
- 競馬自体は2016年1月から始めたので1年と半年ぐらい
  - 競馬は投資の1つとして始めた
- 本業では広く機械学習タスクに携わる

AlphaImpact

- https://alphaimpact.jp/

- 競馬予想界において、ディープインパクトが日本競馬界に与えたような衝撃 (Impact) を、日本で最初 (Alpha) に、人工知能 (AI) によって起こしたいという野望から立ち上げたプロジェクト

- 大元 (@henry0312)
  貫井 (@heartz2001)
  原 (@federalist2015)

- 2017/03/04から netkeiba.com にて予想を提供

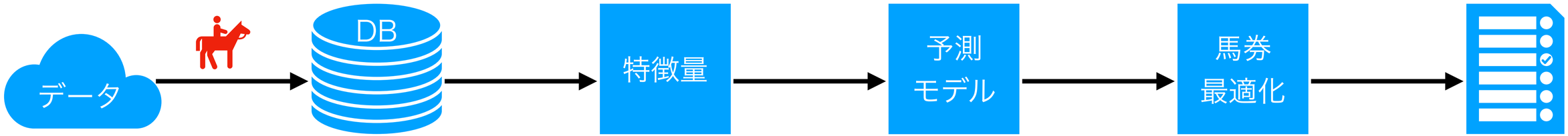
競馬と機械学習

- 機械学習の“Hello World”の次は？
  - MNIST、アヤメの品種、ワインなど
- Kaggleは事前に用意されたデータから正解を導けるかというもので臨場感が薄く，モチベーションの維持が難しい
  - 賞金稼げるぐらいのトップレベルまで到達できるのか？
- 一方、競馬予想は違う

競馬と機械学習

- 毎週解くべき問題となるデータ追加される
- 正解データはレース結果としてリアルタイムに入手できる
- 馬券を買って楽しむことができる
- うまく行けば儲けることができる
- アルゴリズムの試行錯誤 -> 実際に試すのサイクルでモチベーション維持しやすい

競馬予想



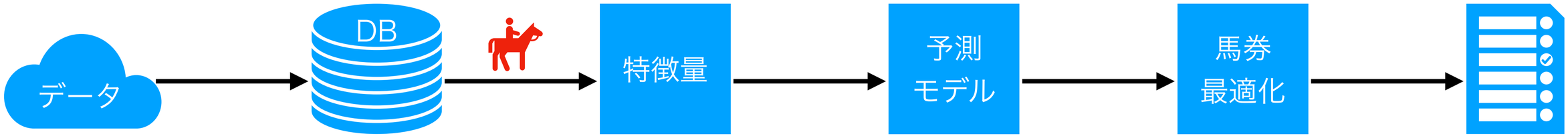
競馬データ

	JRA-VAN DataLab.	netkeiba.com	JRDB
月額料金	2,052円	0〜934円	1,980〜2,480円
開発環境	Windows (Visual Studio)	なんでもOK	なんでもOK
信頼度	◎	○	○
データ種類	○	△	◎
手軽さ	△	◎	○
おすすめの人	• 本格的な運用を考えて居る人 • オッズ解析がしたい人	• 安く手軽に競馬解析したい人	• 趣味〜本格的な運用をしたい人 • 豊富なデータで解析したい人

競馬データ作成時のポイント

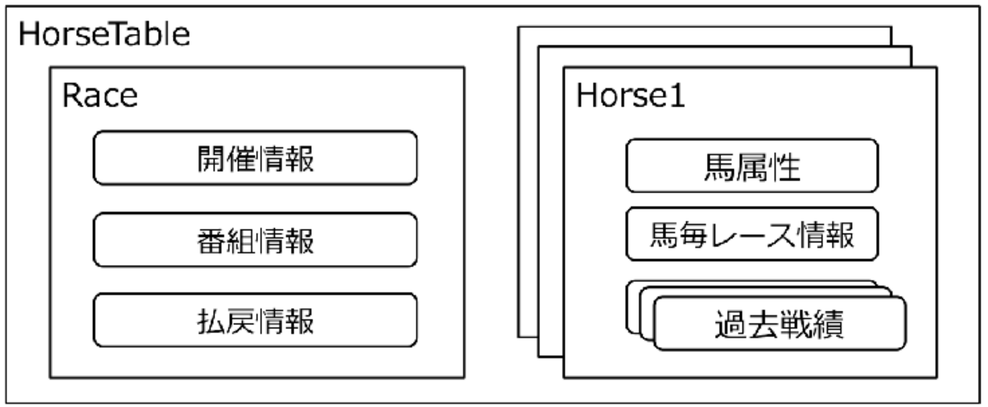
- 欠損値、異常値は当たり前
  - 例えば、坂路以外の調教タイムは人がストップウオッチ使って計測している
- テストが重要
- 型の保証が欲しい
  - 静的型付け言語
  - SQLiteよりはもっと型に厳格なデータベース

競馬予想



特徴量作成の考え方

- オリジナルの馬柱を作る
- 一般の人が利用できない・
  利用していない特徴量を入れる
  - 調教、血統、厩舎など
- 客観的な特徴量を使う
  - 人間の感覚が入ったスピード指数などは利用しない
  - オッズも利用していない



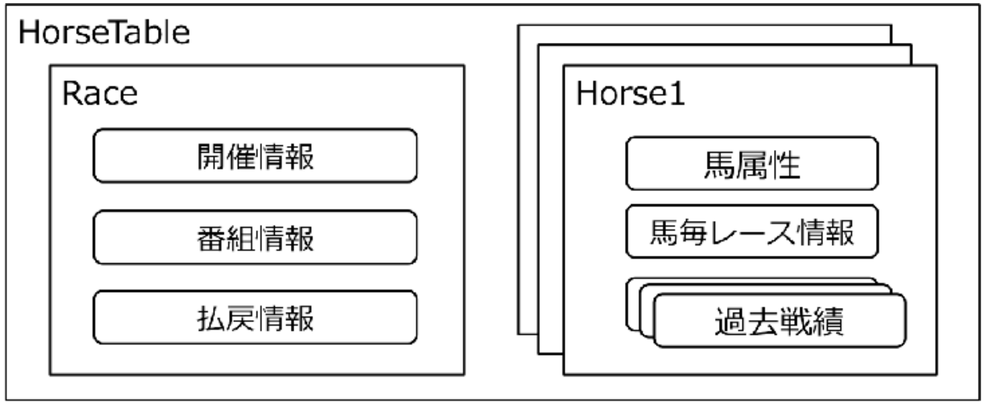
サンプル
⇒ 出走馬

index	性別	馬体重	距離	脚質	連対率
1	1	482	1800	3	0.250
2	2	512	1800	1	0.425
3	1	438	1800	2	0.038
4	3	498	1200	2	0.128

特徴量
⇒ 性別, 馬体重 etc

特徴量作成の悩みどころ

- 過去戦績の扱い
  - 過去何走まで見るのが良いのか？
- 地方・海外での戦績の扱い
- 欠損値の対応



index	性別	馬体重	距離	脚質	連対率
1	1	482	1800	3	0.250
2	2	512	1800	1	0.425
3	1	438	1800	2	0.038
4	3	498	1200	2	0.128

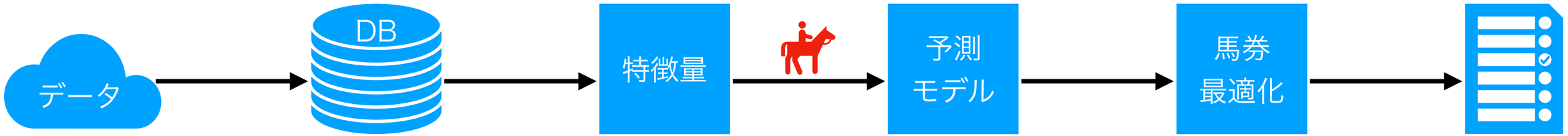
サンプル
⇒ 出走馬

特徴量
⇒ 性別, 馬体重 etc

特徴量作成の注意点

- とにかくテストが重要

競馬予想



何を解くのか？

- 分類
  - 1着になるかどうか
  - 連対するかどうか
  - 複勝圏内にはいるかどうか
- 正例負例の偏りが厳しい
  - ただでさえデータがたくさんあるとは言い難いのに･･･

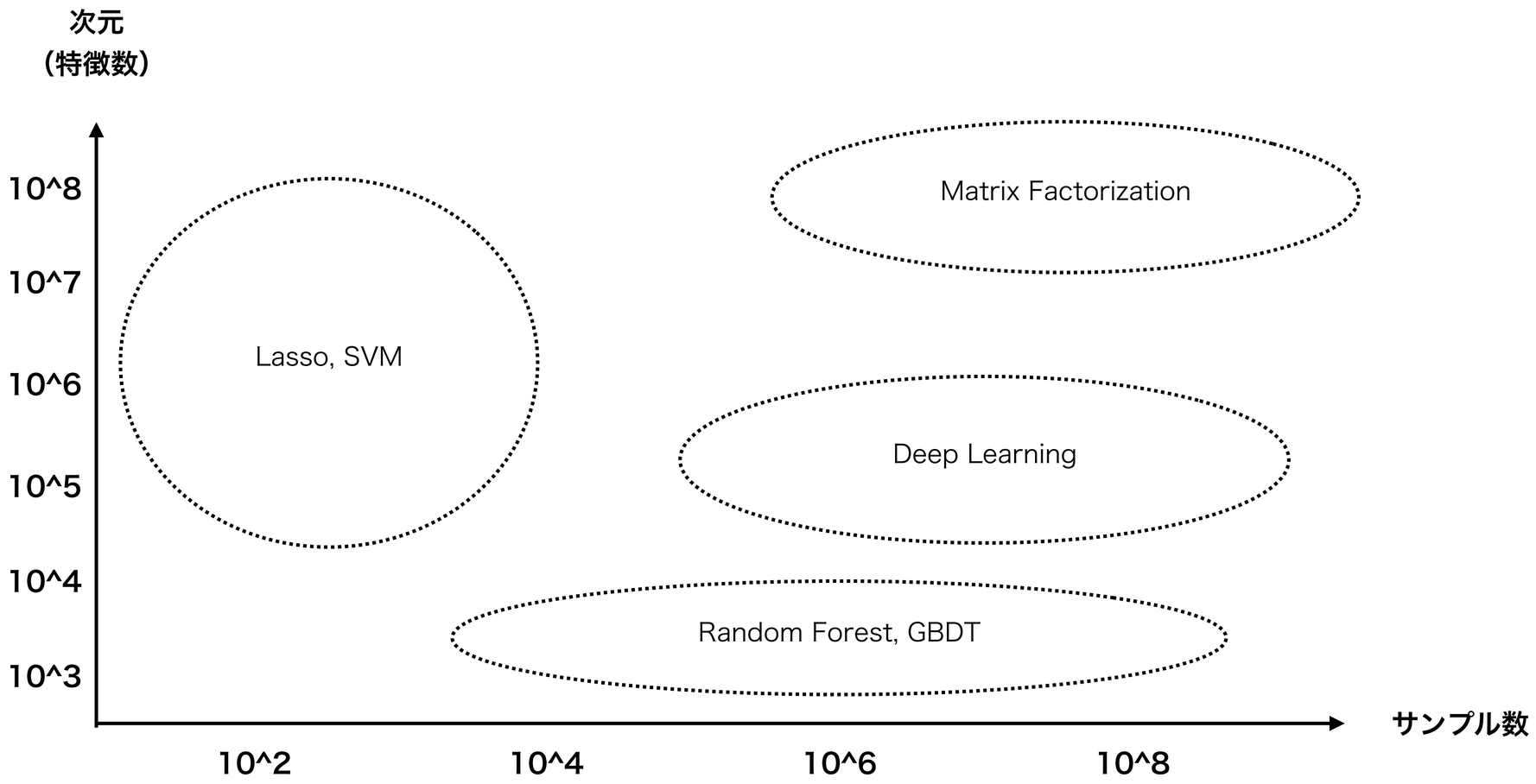
何を解くのか？

- 回帰
  - 走破タイム
  - 着順
  - 獲得賞金
- 分類よりは解きやすい
- 目的変数の設計が肝となる

Learning to Rank

- ランキング学習、ランク学習
  - 情報検索の分野で有名
- 入力データ（出走馬1頭）に対して、そのデータ（馬）が出現する順位（着順）がどれほど価値のあるものかというのを定義して、入力データ（馬）のランキング（着順）を予測する

予測手法の選択



- 図はIBIS2016の山田誠先生の発表『IT企業における機械学習』を参考に一部変更しました

競馬 (AI) の場合

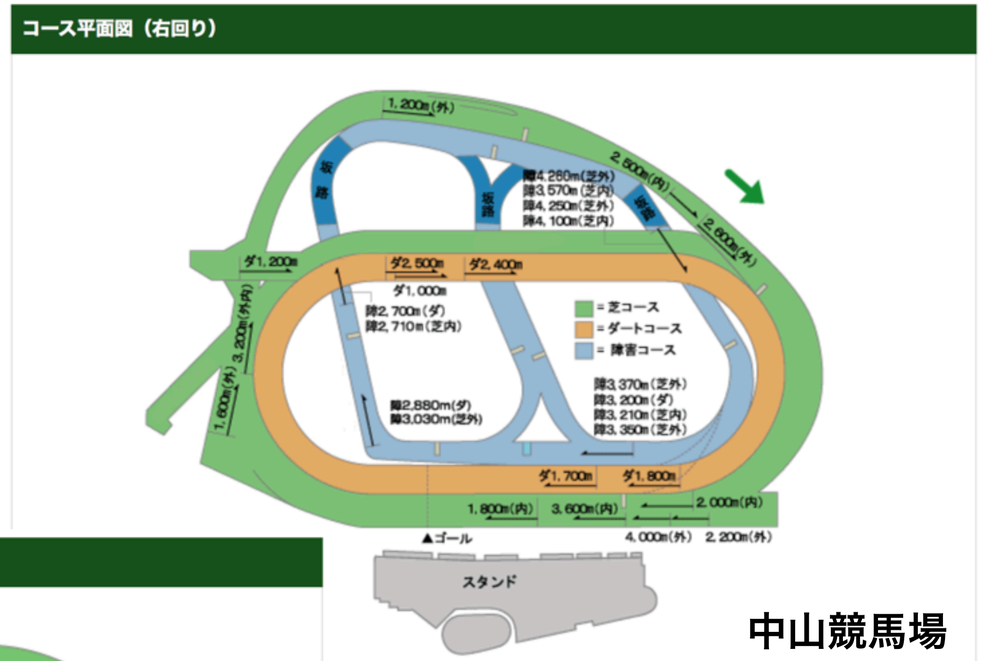
- 特徴数は700ぐらい
  - ダミー化などの処理を入れると最終的に3,000ぐらいの次元数になる
    - 過去のレースをどこまでみるかによっても変わってくる
- サンプル数は1モデル作るのに多くても500レースぐらい
  - 1レース平均14頭出走しているとして7,000サンプル
    - AIはこれを100,000サンプルぐらいまで増やして学習している

AlphaImpactの知見

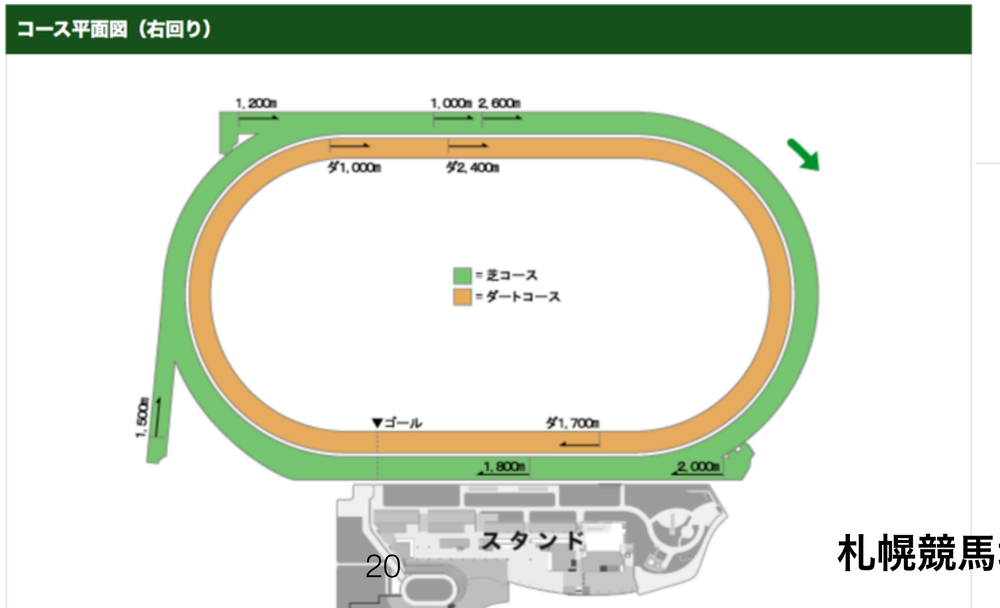
- Deep Learning
  - スパースなデータに向いてない印象
  - コストが大きい割に、言うほど性能が出せない
    - 逆に言えば（時間さえあれば）雑に作ってそこそこ性能が出せる
- Gradient Boosting Decision Tree (GBDT)
  - コストも比較的小さく、性能も良い
  - 非常に過学習が厳しい
  - パラメータチューニングが難しい
- Matrix Factorization
  - 今後、更なる性能向上のキーとなる技術になり得る

データの分割方法

- 芝・ダート・障害
- コース
  - 内回り・外回り
  - 右回り・左回り
- 距離
  - 短距離、中距離、長距離
- クラス
  - 新馬、未勝利、500万下
    1000万下、1600万下
    OP
- 年齢
  - 2歳、3歳、古馬



中山競馬場



札幌競馬場

パラメータチューニング

- 「パラメータチューニング職人の朝は早い」
  - なんて言って良いのは小学生までだよねー
- ベイズ最適化で最適なパラメータを探索する
- 5-fold Cross Validation (5-fold CV) でパラメータを探索する

評価方法

- 訓練データとテストデータを分ける
  - 訓練データ: 2009/01/01 ~ 2016/12/31
    - 中京競馬場のみ、2012/03/01 ~
  - テストデータ: 2017/01/01 ~
- 可能な限り最新のデータを利用してモデルの学習をしたい
- 一方で、テストデータが不足する・・・

評価尺度

- Normalized Discounted Cumulative Gain (nDCG)
  - ランキング予測でよく利用される評価指標
  - よく利用される定義が2つあるので要注意

$$\mathrm{DCG}_p = rel_1 + \sum_{i=2}^{p} \frac{rel_i}{\log_2(i+1)}$$

$$\mathrm{DCG}_p = \sum_{i=1}^{p} \frac{2^{rel_i} - 1}{\log_2(i+1)}$$

- 回収率と的中率
  - nDCGと実際の馬券は必ずしも一致しない

モデル性能/nDCG

東京芝OP

	ベースライン	予測モデル
着順	0.911	0.920
賞金	0.730	0.735
複勝	0.601	0.610
単勝支持率	1.000	0.940

東京芝古馬 1800-2000m

	ベースライン	予測モデル
着順	0.923	0.928
賞金	0.742	0.759
複勝	0.613	0.628
単勝支持率	1.000	0.979

モデル性能/回収率と的中率

東京芝 OP ベースライン

	的中率	回収率	標準偏差
単勝 (Top-1)	0.250	0.575	0.200
複勝 (Top-1)	0.625	0.788	0.196
馬連 (Top-2)	0.000	0.000	NaN
3連単 (Top-3)	0.000	0.000	NaN

東京芝 OP 予測モデル

	的中率	回収率	標準偏差
単勝 (Top-1)	0.375	1.375	1.940
複勝 (Top-1)	0.750	1.087	0.281
馬連 (Top-2)	0.125	0.938	0.000
3連単 (Top-3)	0.125	2.013	0.000

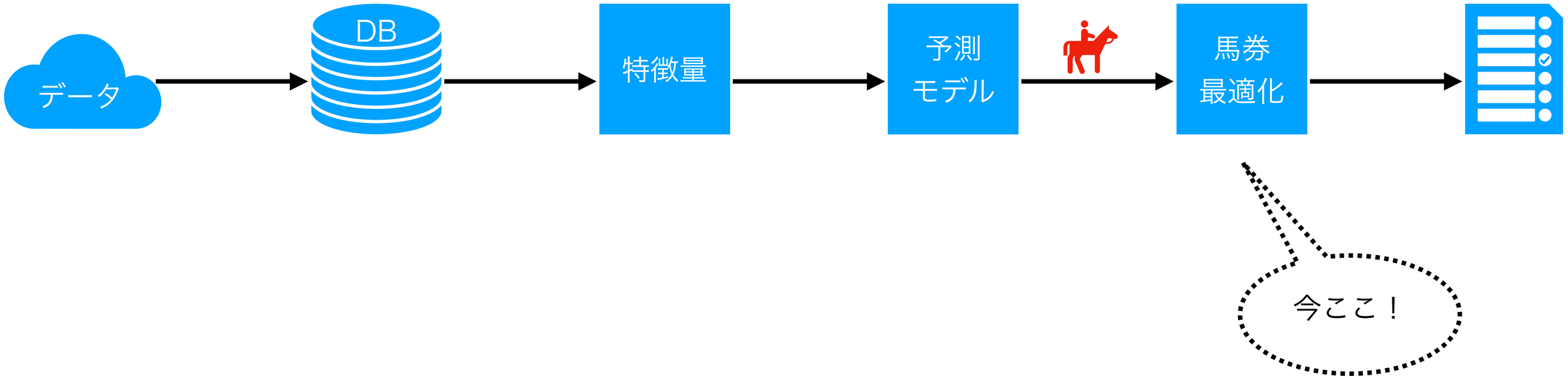
東京芝古馬 1800-2000m ベースライン

	的中率	回収率	標準偏差
単勝 (Top-1)	0.294	0.588	0.276
複勝 (Top-1)	0.647	0.794	0.191
馬連 (Top-2)	0.353	1.965	1.872
3連単 (Top-3)	0.118	0.763	0.233

東京芝古馬 1800-2000m 予測モデル

	的中率	回収率	標準偏差
単勝 (Top-1)	0.353	0.971	1.696
複勝 (Top-1)	0.765	1.065	0.453
馬連 (Top-2)	0.294	1.676	1.364
3連単 (Top-3)	0.235	1.808	4.719

競馬予想



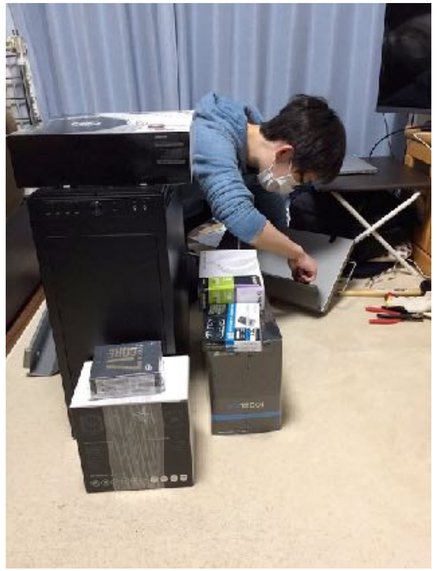
馬券最適化

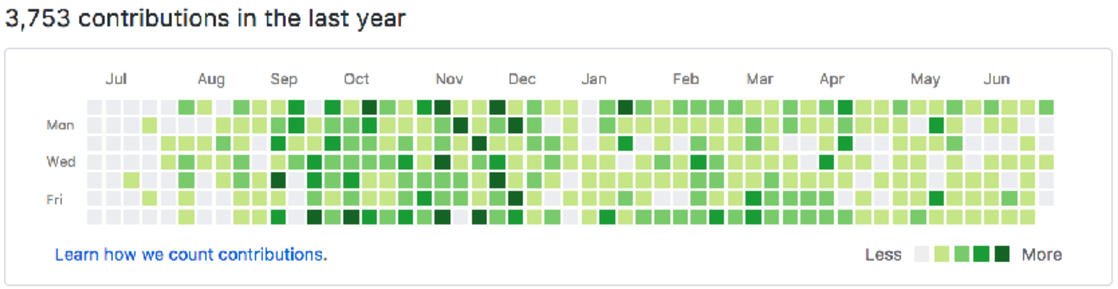
- 100%勝てる予測モデルがあれば全部買えばいいが・・・
- 現実は回収率を高めるためにレースを選択する必要がある
  - 勝負レース
  - オッズ
- 納得のいく方法がまだ見つからず今後の課題
  - 投資戦略まで考慮する必要がありそうではある

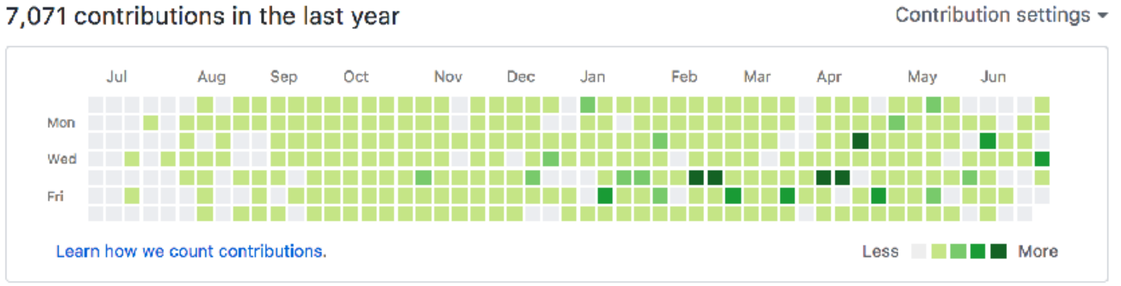
参考情報

AlphaImpactが作成したモジュール

	ソースコード行数	テストケース
データ処理	15,194	476
特徴量	10,5965	3,230
機械学習	16,247	387
(合計)	13,7406	4,093







まとめ

- 競馬のデータソースと取り扱い方
- オリジナルの馬柱を作る感覚で特徴量を作成する
- 問題設定（目的変数の設計）が肝心
- 予測手法はデータの次元数とサンプル数を目安に選択する
- パラメータチューニングは最適化アプローチを使う
- 適切な評価が重要
- 馬券最適化が今後の課題