



Penguin Weight Prediction: Modern ML Serving

～AI協創×サーバーレス×GCP Pro全資格の知見を統合～



【1. DATA & ML LOGIC】 (データの流れ)

- Source: BigQuery Public Dataset
- Pre-process: 欠損値レコードの完全削除によるデータ健全化
- Model: scikit-learn (Regression Pipeline)
- AI Partner: Gemini との共創によるロジック高速生成



【2. ARCHITECTURE】 (インフラ構成図用)



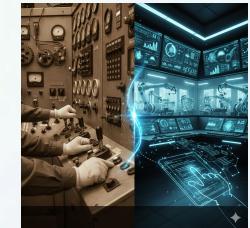
【3. TECH HIGHLIGHTS】 (スキルの証明)

- Efficiency: AIリードエンジニア体制による開発リードタイムの最小化
- Best Practice: Dockerイメージ最適化と環境分離
- Professionalism: GCP Professional 全資格の知見に基づくサービス活用



【4. ROADMAP】 (今後の展望)

- Manual to Auto: CI/CD パイプライン (Cloud Build) への拡張



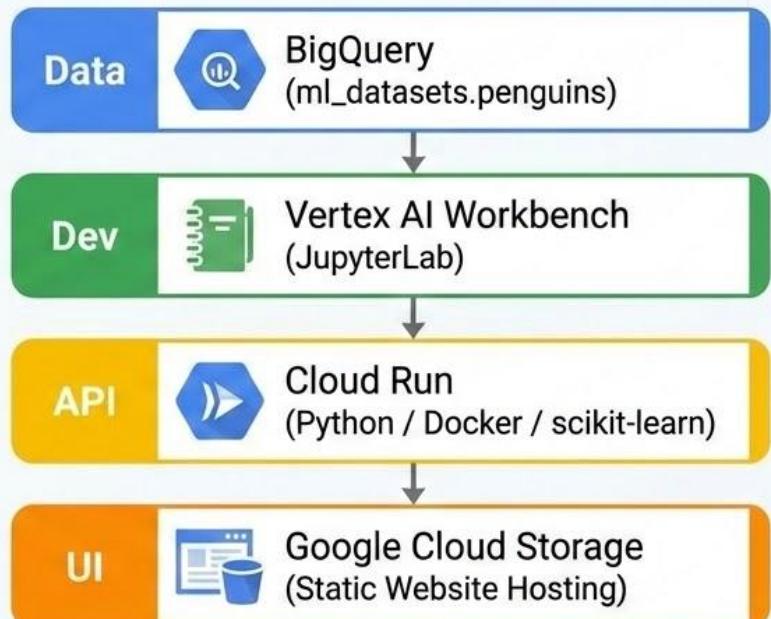
プロジェクト概要とアーキテクチャ設計

～AIとの共創によるサーバーレスML servingの構築～

プロジェクトの目的

- BigQuery公開データセットを用いた、実用的な機械学習デモのWeb公開。
- AI (Gemini) を共同開発パートナーとし、構想からデプロイまでのリードタイムを最小化する手法の検証。

システムアーキテクチャ



データ戦略とAI協創プロセス

～データの健全性確保とモデルの検証～

データエンジニアリング



データセットの項目



ペンギン種



生息する島



体の特徴（くちばしや羽の長さ）



性別



体重



目的量 → 体重



特徴量 → 体重以外の項目



欠損値処理 予測精度の安定性を優先し、欠損値を含むレコードを全て削除。

AIとの共同開発フロー



ロジック
生成

Geminiに分析要件を伝え、scikit-learnを用いた回帰パイプラインのベースコードを生成。



UI生成

Geminiにデザイン要件を伝え、ベースのHTMLを生成。モバイルでの操作性を意識して手動で修正。



検証

生成された前処理（One-Hot Encoding等）とアルゴリズムの妥当性を、Vertex AI Workbench上で検証。



成果

「AIによる高速生成」と「人間による技術検証」のループにより、信頼性の高いロジックを短期間で確立。

サーバーレス環境へのデプロイと今後の展望



■Container-Based Deployment

- Cloud Runへのデプロイ: python:3.10-slim をベースとした軽量コンテナを採用。
- ビルドの最適化: --no-cache-dir の適用など、コンテナの仕組みを活かしたイメージ構成。
- AI協創プロセス: Geminiが提案したDockerfileを使用し、実際のデプロイを通じた動作確認を実施。



■Service Integration & Accessibility

- 公開設定: デモとしての利便性を優先し、Cloud RunおよびGCSバケットをパブリックアクセス(allUsers)に設定。
- 疎通確認: GCSホスティングされたUI(HTML)からCloud Run APIへのシンプルな呼び出しフローを実現。



■Roadmap: Workbenchから Opsフローへの拡張

- 現状: Workbench上での開発および手動デプロイを実装。
- Next Action: 手動オペレーションを脱却し、Cloud Build等を活用したCI/CDパイプラインによる自動デプロイの構築。

(まとめ) 実装者の理解度と成果

～リクルーターへのメッセージ～

GCPのエコシステム理解

- データの抽出からWeb公開まで、最適なサービスを選択・統合する能力。

モダンな開発手法

- 生成AI (Gemini) をリードエンジニアとして使いこなし、開発効率を最大化する「AI協創型」のワークフロー。

実務的な技術基盤

- GCP Professional全資格取得により得た知識を活用し、BigQuery/WorkbenchによるML開発、Dockerによる環境構築、アクセス権限設定を実装。
- また過去の開発経験を生かしたUIデザイン(HTML)の作成。