

RAJASHEKAR REDDY VEDIRE

Indianapolis, IN | rvedire.com | linkedin.com/in/rvedire | github.com/Prodoorknob
MS Applied Data Science, Indiana University Indianapolis (Aug 2026) | F1 Visa, STEM OPT Eligible

Summary

Applied scientist with industry experience in anomaly detection (Microsoft Research) and data analysis (Microsoft Advertising), currently completing an MS in Applied Data Science. Building production ML systems across cricket analytics (Bayesian win probability, ILP optimization, tabular Q-learning), agricultural forecasting (SARIMAX + LightGBM ensemble with SHAP explainability), healthcare cost prediction (103M-row national-scale pipeline), and autonomous web agents (vision-language model perception). Track record of shipping end-to-end: from data pipeline through model training to deployed API with caching, background scheduling, and monitoring. Active work in LLM agent architectures, multi-modal VLM systems, and Bayesian methods.

Education

MS, Applied Data Science Aug 2024 – Aug 2026
Indiana University Indianapolis (Luddy School) Indianapolis, IN

- Graduate Research Assistant, Sport Innovation Institute (SII)
- Coursework: Machine Learning, Applied Deep Learning, NLP, Statistical Computing, Data Mining, Big Data Analytics, Cloud Computing

B.Tech, Mechanical Engineering (Automotive) 2013 – 2017
Vellore Institute of Technology India

Technical Skills

Machine Learning / AI: Logistic regression, isotonic calibration, Bayesian inference (Beta priors, log-odds updates), Monte Carlo simulation, LightGBM, XGBoost, Random Forest, SARIMAX, Holt-Winters exponential smoothing, tabular Q-learning (MDP), integer linear programming (PuLP/CBC), mixed-effects regression, TF-IDF + SVM text classification, shrinkage estimation, SHAP explainability, walk-forward cross-validation, quantile regression, Mahalanobis anomaly detection

LLM / Multi-Modal: LangGraph agent orchestration, RAG pipelines (FAISS + Ollama embeddings), vision-language model integration (Qwen3-VL, Granite Vision), Anthropic API (narrative generation), prompt engineering for structured output, multi-turn state persistence

Languages: Python (primary), TypeScript, SQL

Frameworks: FastAPI, Next.js (App Router), Streamlit, Playwright, scikit-learn, pandas, NumPy, SciPy, statsmodels, PyArrow, Pydantic v2, SQLAlchemy 2.0 (async), Alembic, Recharts, Deck.gl

Data / Cloud: PostgreSQL (Supabase, AWS RDS, Azure SQL), Redis, DuckDB, BigQuery, Databricks/Delta Lake, PySpark, AWS (S3, EC2, Athena, SNS, EventBridge), Azure (Static Web Apps, Entra ID, Managed Identity), GCP (Dataproc, Cloud Scheduler)

Infrastructure: Docker (NVIDIA CUDA containers), GitHub Actions CI/CD, Railway, Vercel, APScheduler, systemd services, Alembic migrations, MLflow experiment tracking

Visualization: Power BI, Plotly, Chart.js, Recharts, Deck.gl + MapLibre GL (choropleth maps), html-to-image (social export)

Experience

Graduate Research Assistant*Sport Innovation Institute, Indiana University*

Aug 2024 – Present

Indianapolis, IN

- Designed and deployed the Horizon League Budget Dashboard: migrated from Power BI (\$250/mo licensing) to a custom Next.js application on Azure Static Web Apps (\$59/mo), serving 11 NCAA Division I Athletic Directors with 8 analytics pages, Entra ID multi-tenant authentication, and Azure SQL star schema (11 dimensions, 7 facts, 670+ salary records)
- Architected DataSkive cohort analytics on GCP/BigQuery: reverse-engineered a partially documented database (30+ tables, 250+ columns), discovered that behavioral segment is 114× more predictive of conversion than cohort assignment, identified \$857M in attributed revenue across 48 cohort profiles, and built a 3-tier anomaly detection pipeline (Holt-Winters, Prophet, LSTM) instrumenting 5 KPI signals
- Built Intent Quotient (IQ): original NLP metric quantifying batting aggression intent from ESPNericinfo commentary using TF-IDF + SVM classification (40+ attack pattern regexes, calibrated probability output), with Bayesian shrinkage regularization against season-wide player priors for phase-stratified analysis across 750+ IPL matches (2007–2024)
- Engineered IPL Playing XI Selector: 4-layer prescriptive pipeline combining PuLP ILP with hard constraints (overseas cap, bowling coverage, role minimums), mixed-effects synergy regression from cross-league partnership data (BBL/CPL/PSL/SA20/T20WC), and tabular Q-learning MDP with 384 discretized game states and SR-delta reward shaping; achieved 0.602 Jaccard similarity on 148-match backtest with formal divergence taxonomy
- Developed PlayerData athlete benchmark system: data engineering pipeline for 7,868 wearable sessions across 281 collegiate athletes, percentile computation by cohort (sport × gender × division × age band), rule-based conversational chatbot with intent classification and entity extraction for natural language benchmark queries

Programmer Analyst, Client: Microsoft Research*Cognizant Technology Solutions*

Aug 2021 – May 2024

Bangalore, India

- Evaluated multiple time series prediction algorithms for anomaly detection, comparing feasibility and performance across Microsoft Research engagement metrics
- Implemented a modified Holt-Winters algorithm with forward testing and back testing in Databricks for performance analysis
- Achieved 45% reduction in false positives at user-level with no compromise in recall

Programmer Analyst, Client: Microsoft Advertising*Cognizant Technology Solutions*

Bangalore, India

- Performed data analysis across Microsoft Ads products using internal tools (Agora, Scope, PyScope) and SQL/Python
- Designed and developed business-specific Power BI dashboards saving 20–30% time in operational procedures
- Integrated Power BI data pipelines with SQL and Python for comprehensive business requirement delivery and automation

Selected Projects**CoverDrive Cricket**

Mar 2026 – Present

Full-stack IPL analytics platform — FastAPI, Next.js 16, PostgreSQL, Redis, scikit-learn, Anthropic API

- Built 3-stage Bayesian win probability engine: Stage 1 logistic regression with isotonic calibration (Brier = 0.194 on held-out 2024 data), Stage 2 Bayesian log-odds update incorporating toss advantage and playing XI quality delta, Stage 3 NumPy-vectorized Monte Carlo engine (10K simulations, p99 latency)

47ms) for ball-by-ball live probability updates

- Designed context-adjusted performance metrics (SR+, Avg+, Eco+) using 4-factor decomposition: era (2023 Impact Player Rule boundary), venue, match phase (powerplay/middle/death), and opposition quality; computed from 2M+ ball-by-ball delivery records
- Implemented shrinkage matchup model blending empirical batter-vs-bowler records with Laplace-smoothed 5×2 prior matrix (bowler type \times batter handedness), enforcing minimum 20% prior weight to regularize small-sample estimates
- Built EWA form indicators (exponentially weighted average, $\alpha=0.3$) and team composition metrics (Batting Composition Index, bowling depth, allrounder ratio) feeding into win probability as structural features
- Engineered match preview caching system: stale-while-revalidate pattern with Redis, cold preview builds reduced from 88 to 23 database queries via N+1 batch optimization, warm reads served from single Redis GET
- Integrated SportMonks Cricket API for live scoring (30s polling during match hours), Claude narrative pipeline for AI-generated match analysis (8–10 typed bullets per match with manual review gate), and Razorpay freemium monetization with Clerk authentication
- Initiated Phase A cross-league lambda coefficient estimation: Bayesian approach with Beta priors estimating 30 league quality coefficients across 6 T20 leagues using player-overlap evidence for IPL-equivalent normalization

Agricultural Data Analysis (QuickStats)

Nov 2025 – Present

USDA analytics dashboard with commodity price forecasting — Next.js, FastAPI, LightGBM, SARIMAX, AWS

- Built end-to-end commodity price forecasting module: ensemble model (SARIMAX for seasonality + LightGBM with quantile regression for probabilistic spread + Ridge meta-learner + isotonic calibration) producing p10/p50/p90 forecasts for corn, soybeans, and wheat across 1–6 month horizons
- Engineered 18-feature matrix from 4 data sources: market signals (CME futures, term spread, basis, open interest), fundamentals (WASDE stocks-to-use ratio and percentile, surprise magnitude), macro (DXY level and 30-day change), cost metrics (ERS production cost per bushel, price-cost ratio); all validated with Pandera schema enforcement
- Implemented Mahalanobis distance regime anomaly detection at inference time: flags out-of-distribution feature vectors and defers to futures curve rather than trusting model predictions during regime shifts
- Applied SHAP TreeExplainer for per-forecast key driver identification, surfacing top contributing feature (e.g., “stocks-to-use percentile” or “futures basis spread”) as interpretable callout in dashboard
- Designed walk-forward validation pipeline (2010–2019 train, 2020–2022 validation, 2023–2024 test) with futures-baseline MAPE gate (model must beat naive futures + 1.5 percentage points)
- Migrated frontend from Streamlit to Next.js 16 with browser-side parquet reading (hyparquet), Deck.gl choropleth maps, and S3-first data fetching; total infrastructure cost \$22/month (RDS \$15, EC2 \$6, S3/Athena <\$1)

DataSkrive Cohort Analytics

Mar 2026 – Present

Cohort audit and anomaly detection for sports betting content platform — BigQuery, PySpark, statsmodels, LangGraph

- Reverse-engineered partially documented GCP/BigQuery database: mapped 30+ tables and 250+ columns into a living schema inventory with confidence levels, discovered two coexisting cohort architectures (static A/B experiment groups vs. dynamic content-variant targeting)
- Quantified that behavioral segment is $114 \times$ more predictive of conversion than cohort assignment using BQML Boosted Tree feature importance (AUC 0.885, 194M rows); identified scenario selection as largest

optimization lever (183× conversion rate spread across 19 active scenarios)

- Built Holt-Winters anomaly detection pipeline: additive seasonal decomposition with 7-day period, grid-search parameter tuning (α , β , γ), bootstrap training window, 3-point moving average residual scoring with percentile bounds; validated 4 anomaly signals including system-wide CVR collapse (0.25 → 0.08) and \$267K NBA halftime value destruction
- Constructed local RAG pipeline: LangGraph 6-node state machine with FAISS vector index over 75KB+ schema documentation, Ollama Qwen3-14B/32B for generation, SQLite state persistence for multi-turn conversations; deployed via Streamlit web interface for team knowledge access
- Proposed 5 concrete reclassification strategies with SQL implementation sketches; documented weight normalization bug in active cohorts (Cohort 1 weight sum 3.29 vs. expected 1.0)

Medicare Provider Cost Analysis

Mar 2026 – Present

National-scale ML pipeline for CMS data — scikit-learn, XGBoost, RAPIDS cuML, MLflow, Databricks

- Built medallion pipeline (Bronze → Silver → Gold) processing 103M rows of CMS Physician & Practitioners data (2013–2023) with dual execution modes: Databricks/PySpark for production, pandas/PyArrow for local development with per-state parquet partitioning
- Implemented regional batch training by Census region to manage memory: XGBoost with booster continuation (125 rounds/region), Random Forest with warm_start (125 trees/region, 625 total), CUDA auto-detection for GPU acceleration on RTX 5070 Ti
- Achieved $R^2 = 0.884$ (Random Forest, test MAE \$12.04) after removing data leakage: identified and excluded payment-derived features (Avg_Mdcr_Pymt_Amt, standardized amount, ratio features) that had inflated initial R^2 to 0.9996
- Integrated provider-level HCC risk scores from separate CMS dataset via NPI+year join (median imputation for missing values); engineered 10-feature set including clinical HCPCS bucketing (6 categories), log-transformed volume metrics, and facility/office flag
- Prepared LSTM time-series sequences: 23,672 provider-service-state groups with year-ordered target vectors (10,540 complete 11-year series) for Phase 3 temporal forecasting

Peruse AI (Open Source)

Feb 2026

Autonomous VLM web agent — Python, Playwright, Ollama, LM Studio, Jina

- Published local-first autonomous web agent to PyPI: perceive-plan-act loop combining dual-channel perception (DOM element extraction + visual screenshots) with vision-language model decision-making via Playwright browser automation
- Implemented 5-strategy VLM response parsing fallback for handling malformed JSON from local models: markdown stripping, direct parse, brace-matching, partial extraction, and natural language keyword recovery; agent falls back to scrolling on complete parse failure to maintain exploration continuity
- Built concurrent focus group execution: multiple personas (UX designer, accessibility auditor, data analyst) running in parallel against the same URL, each with independent browser instance, VLM session, and output directory; generates structured Data Insights, UX Review, and Bug Report outputs
- Designed smart loop recovery with nudge messages: detects identical consecutive actions (7+ repeats) and low-variety oscillation (2 unique actions over 12 steps), issues progressive nudges with element blocking to prevent re-interaction

APEX (Algorithmic Trading System)

Mar 2026

Multi-signal ensemble framework for US equities — Python, DuckDB, LightGBM, Alpaca, FRED, SEC EDGAR

- Designed 8-layer pipeline architecture: raw data sources (Alpaca, FRED, SEC EDGAR, Finnhub) → ingestion engine → signal engineering (47 features across 8 domains) → DuckDB feature store (15-table

star schema with point-in-time correctness) → regime classifier → prediction ensemble → risk gate → execution

- Built 5 production ingestors following BaseIngestor pattern (fetch/transform/store with rate limiting and INSERT OR REPLACE deduplication): 1-minute OHLCV via Alpaca, 5 macro series via FRED, SEC EDGAR 8-K filings and Form 4 insider trades with XML parsing, financial news via Finnhub, yfinance as price fallback
- Implemented 9 sequential circuit breaker risk gates: minimum confidence ($|p_{up} - p_{down}| \geq 15\%$), maximum position ($\leq 25\%$ capital), sector concentration ($\leq 40\%$), correlated positions (≤ 3 with $\rho > 0.7$), daily loss halt (5%), drawdown halt (15%), VIX circuit breaker ($VIX > 35 \rightarrow 50\%$ reduction), liquidity ($< 1\%$ ADV), losing streak dampening (3 consecutive $\rightarrow 50\%$ weight)
- Built half-Kelly position sizing with calibrated probability inputs and TFT quantile support; vectorbt backtesting engine computing Sharpe, max drawdown, win rate, profit factor, and Calmar ratio

Ball View

Feb 2026

Real-time cricket ball tracking for live broadcasts — FastAPI, YOLOv8, OpenCV, Kalman Filter, EasyOCR, Docker/CUDA

- Built real-time computer vision pipeline: browser-side video capture (Chrome Extension + Tampermonkey) at 30 FPS via WebSocket, YOLOv8n object detection (confidence threshold 0.3 for small cricket ball recall), Kalman Filter trajectory tracking (constant velocity model, 20-frame history), HSV pitch detection for scene classification and spatial gating
- Implemented OCR-to-match-data synchronization: EasyOCR reads scoreboard region every 30 frames, fuzzy-matches team names and over counts against Cricsheet ball-by-ball JSON to retrieve live batter, bowler, and run context without a separate data feed
- Deployed on NVIDIA Docker (CUDA 12.4) with PowerShell launcher for WSL2 GPU passthrough on RTX 5070 Ti

Horizon League Budget Dashboard

Jan – Feb 2026

Athletic budget benchmarking for 11 NCAA D1 universities — Next.js, Azure SQL, Entra ID, GitHub Actions

- Replaced Power BI dashboard (\$250/mo licensing for 20 users) with custom Next.js application on Azure Static Web Apps (\$59/mo): 8 analytics pages (league overview, basketball deep dive, sport-by-sport, salaries, NIL/COA, department operations, facilities, year-over-year trends) with Recharts and TanStack React Table
- Designed Azure SQL star schema (11 dimensions, 7 facts) with Python ETL for Qualtrics survey responses and EADA federal reporting data; implemented Azure Managed Identity for credential-free database access and multi-tenant Entra ID authentication supporting B2B guest accounts from 11 institutional Azure AD tenants

Open Export (Open Source)

Feb 2026

ChatGPT conversation archival tool — Python, Playwright, Click, Rich

- Published CLI tool to PyPI for automated ChatGPT conversation export: connects via Chrome DevTools Protocol, paginates conversation list API, linearizes tree-based message structures into chronological threads, exports to JSON + Markdown with SHA-256 deduplication

PocketLedger

Mar 2026

Local-first personal finance tracker — FastAPI, Tesseract OCR, OpenCV, Ollama, SQLite

- Built OCR-powered bank statement parser with auto-detecting bank routing (Chase, BofA, Discover), image preprocessing pipeline (grayscale \rightarrow denoise \rightarrow adaptive threshold \rightarrow deskew), and dual categorization engine (SQL pattern matching + Ollama LLM fallback with source tracking); supports 6 CSV

bank formats with heuristic auto-detection