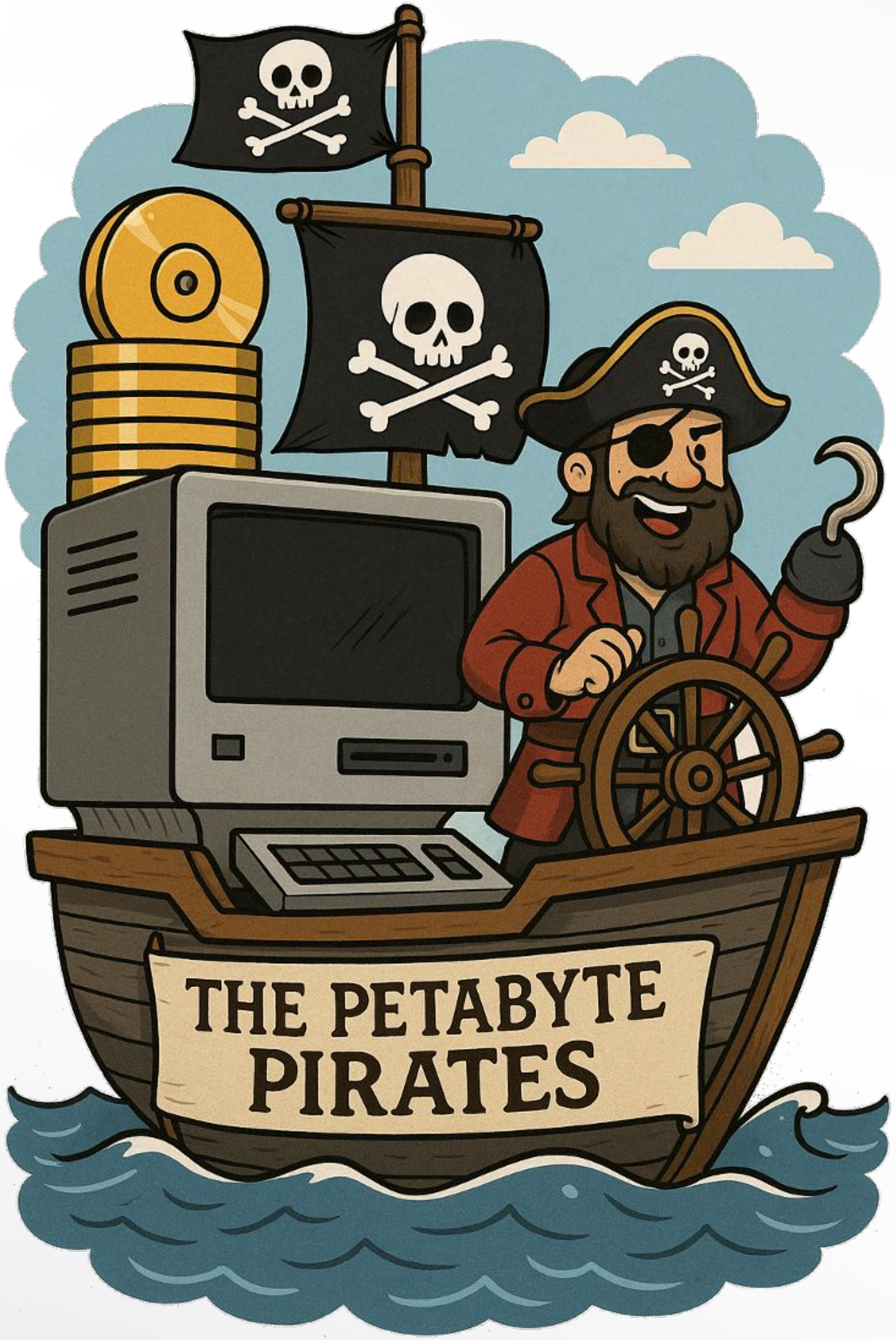
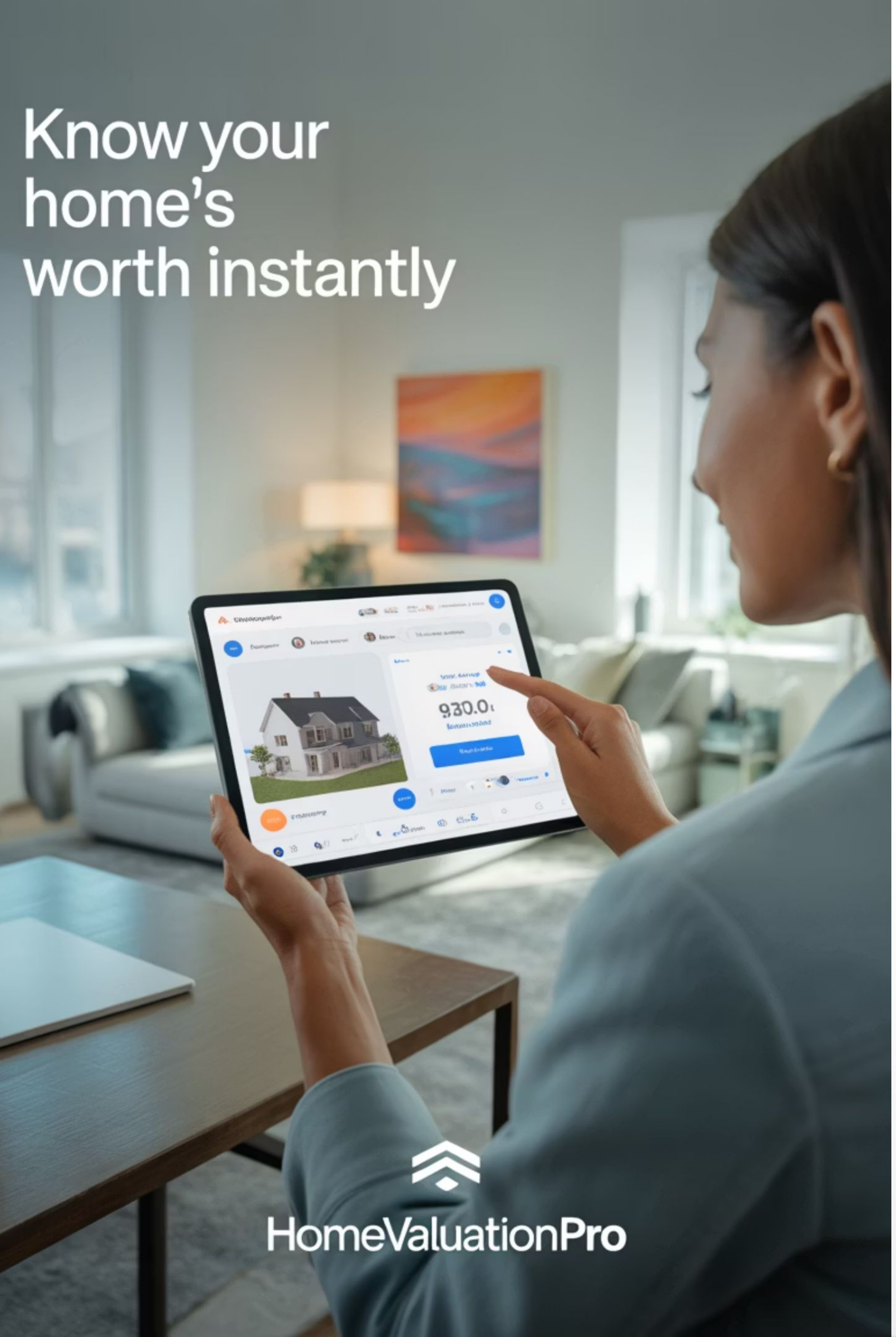
THE PETABYTE PIRATES - SIGNING OFF

Automated Property Valuation: From Model to Market

ACS-5513 Applied Machine Learning - Checkpoint 3

Team: Sean Miller, Farhan Hassan, Dewayne Hafenstein





Executive Summary & Key Insights

What We Built

Full-stack ML service predicting single-family home prices in Ames, Iowa with a production-ready API and web interface for instant valuations

68.64% of predictions within ±10% of actual sale price

Key Business Insights

- Quality × Size interaction explains 80% of price variance alone
- Recent renovations add $15-25K premium vs. original construction
- Garage presence increases home value by $26K on average
- Premium neighborhoods command $70-90K premiums

Business Impact

- Reduce appraisal time from days to seconds
- Eliminate human bias in property valuation
- Save thousands of dollars per transaction in mispricing risk

Industry Context & Benchmarking

The Automated Valuation Models (AVM) landscape is dominated by major players, but our solution offers competitive advantages in a specialized market.

Platform	Market Share	Accuracy	Coverage
Zillow Zestimate	40%+	1.9% median error	National
Redfin Estimate	25%+	2.2% median error	Major markets
CoreLogic AVM	20%+	Industry standard	Commercial
Our Model	Prototype	**6.38% median error**	Ames, IA

Competitive Positioning

Zillow Zestimate: 1.9% median error, but limited feature transparency

Redfin Estimate: 2.2% median error, better explainability

Our Advantage: Full feature transparency + domain expertise + local specialization

Regulatory Environment

Freddie Mac ACE+ PDR: Allows appraisal waivers for AVMs meeting accuracy thresholds

Our Target: <10% error rate enables regulatory compliance pathway

Success Metrics: Absolute vs. Relative Performance

Why Both Dollar and Percentage Metrics Matter

Absolute RMSE
$20,708
Mortgage underwriting context: "Typical miss is ~$21K"

Relative Error
6.38% median
Real estate standards context: "Usually within 6% of true value"

Precision Bands
68.64% within ±10%
Appraisal variance context: "More accurate than human appraisers"

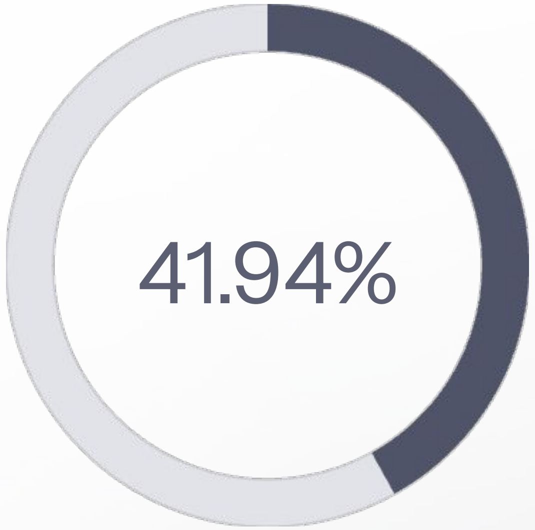
Ames Market Context

- Median home price: ~$180K (2006-2010 data)

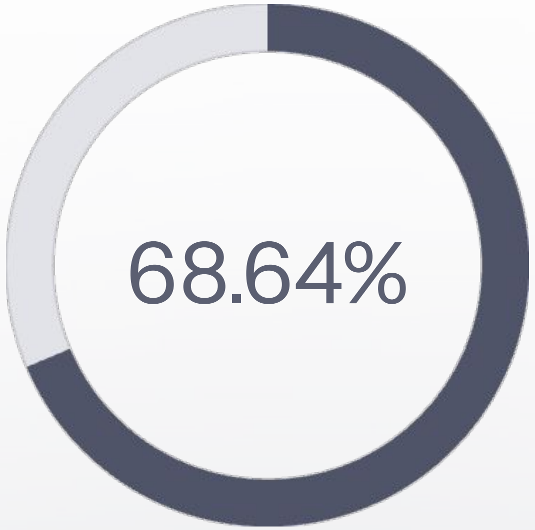
$20K RMSE = **11.5%** of median price

- Industry standard: 10-15% acceptable for lending decisions

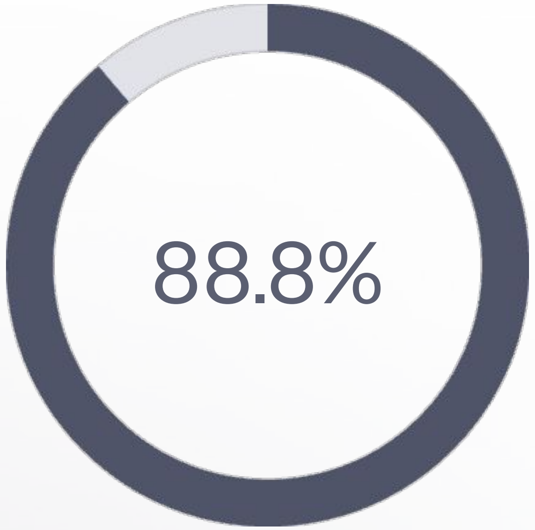
Performance Distribution



Homes within ±5% (excellent predictions)



Homes within ±10% (acceptable for lending)



R^2 explains nearly 9/10 of price variance

Data Foundation & Feature Engineering

Dataset Characteristics

Source: Ames Housing Dataset (Kaggle)

Size: 2,930 transactions → 2,789 after cleaning

Features: 81 raw → 30 engineered final features

Time Period: 2006-2010 (pre-financial crisis)

Data Quality Pipeline

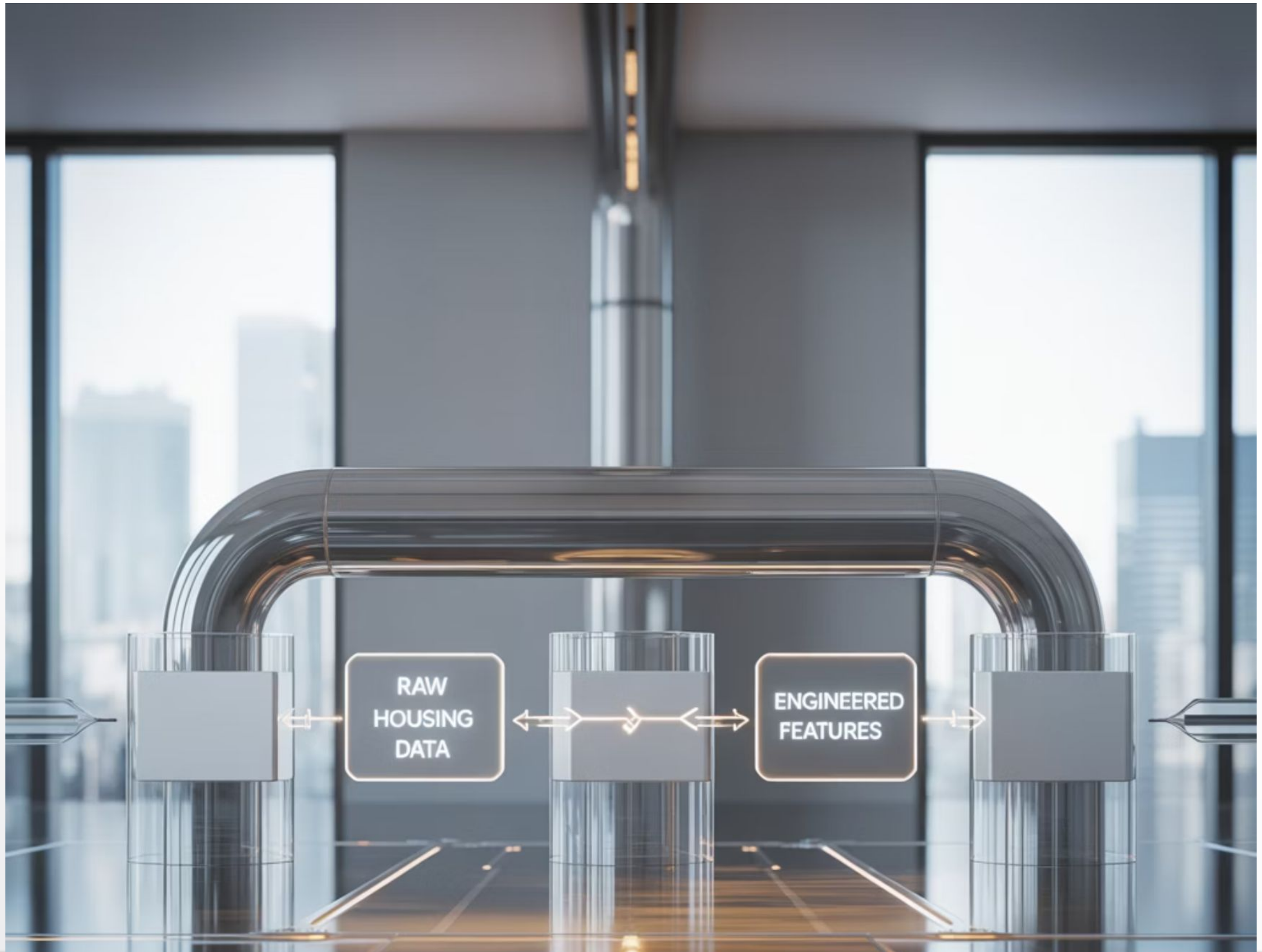
Outlier Removal: 1.5×IQR rule (137 extreme values removed)

Missing Data: <5% after feature selection

Multicollinearity: VIF < 5 for all final features

Feature Engineering Breakthroughs

Feature	Correlation	Business Logic
Qual × SF Plus Garage + Garage Finish × Garage Area	**0.906**	Quality amplifies size value, garage adds premium
Total SF Plus Garage	0.82	Living space + storage drives value
Overall Quality	0.79	Construction quality is primary driver
House Age	-0.59	Newer homes command premium



Feature Correlation Analysis

Candidates for Feature Engineering

Key Correlation Findings

Highest Positive: Qual×SF interactions (0.90+)

Highest Negative: Age-related features (-0.56 to -0.59)

Surprising Weak: Overall Condition (0.06) - condition ≠ quality

Feature Reduction: 107 → 30 features with minimal R^2 loss

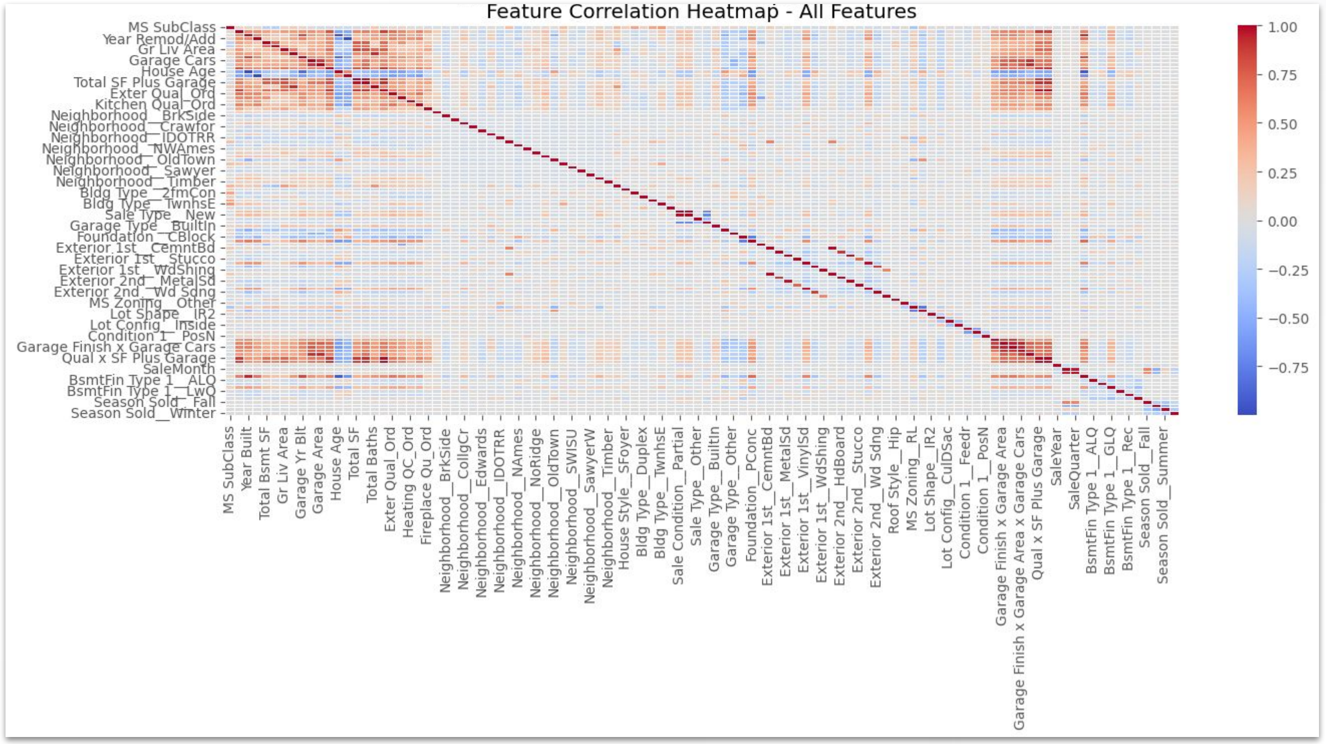
Multicollinearity Management

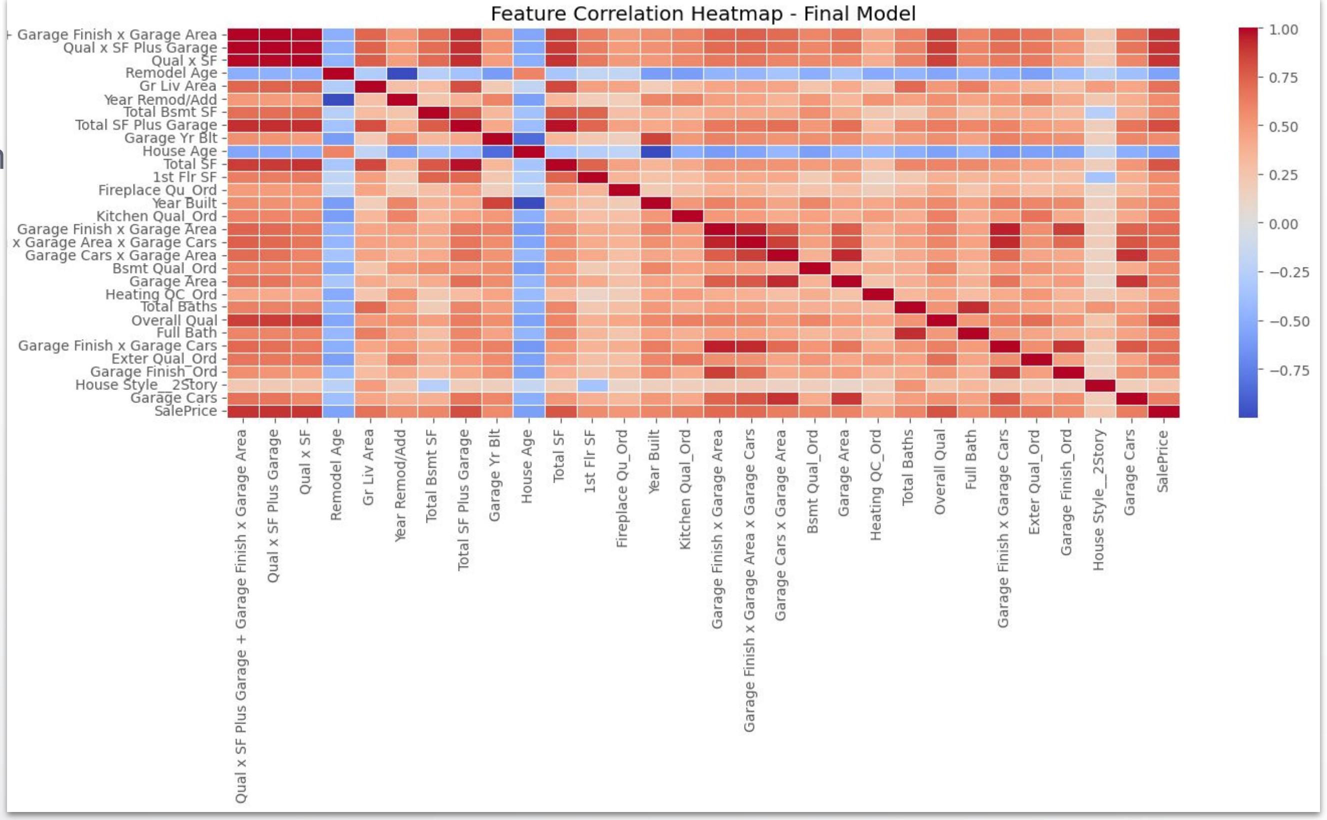
Pre-selection

|correlation| > 0.5 with target

Post-selection

Engineered first order interaction terms, particularly with quality ordinals.





The final feature set contains intentional multicollinearity (e.g., Qual × SF and its raw inputs) to preserve interpretability for stakeholders. **Tree-based algorithms are insensitive to collinearity**; for linear baselines we used Ridge/Lasso regularisation.

Model Development & Hyperparameter Tuning

Model Zoo Comparison (Updated)

Model	RMSE	R^2	Deployment
CatBoost (Tuned)	**$20,708**	**0.888**	✅ Production
Stacking Ensemble	$20,142	0.872	Complex
XGBoost	$20,592	0.866	Alternative
Random Forest	$20,632	0.865	Baseline

CatBoost Hyperparameter Optimization

Optimization Framework: Optuna with 100 trials

Search Space:

- Iterations: 100-2000
- Learning Rate: 0.01-0.3
- Depth: 3-10
- L2 Regularization: 1-10

Final Parameters: 1037 iterations, 0.027 learning rate, depth 5

Why CatBoost for Production?

Native categorical handling

No preprocessing needed

Robust to overfitting

Built-in regularization

Fast inference

<300ms response time

Model interpretability

SHAP integration

Cross-Validation & Model Stability

5-Fold Cross-Validation Results

Fold	RMSE	R^2	Performance
Fold 1	$20,614	0.871	Consistent
Fold 2	$19,428	0.875	Best performing
Fold 3	$20,638	0.881	Consistent
Fold 4	$22,380	0.849	Acceptable variance
Fold 5	$19,945	0.891	Excellent

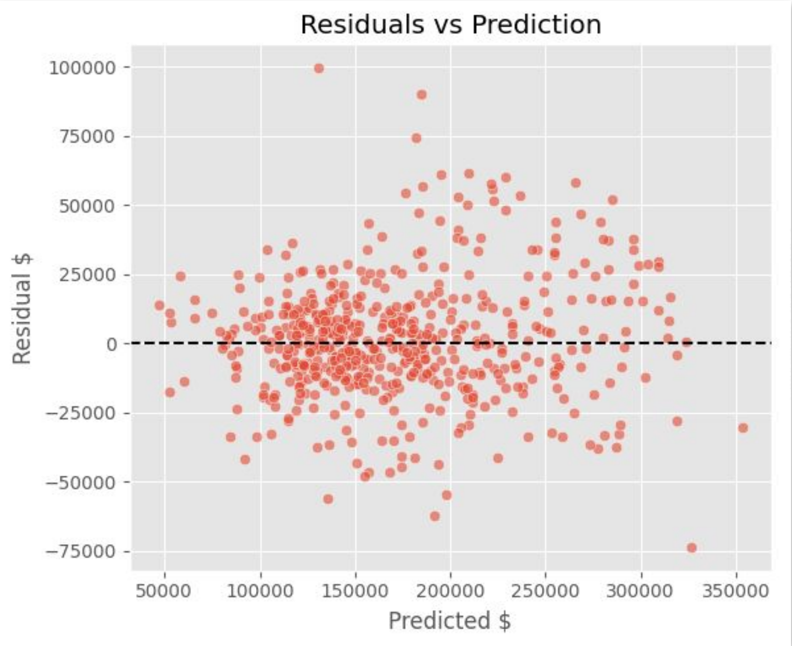
Statistical Stability

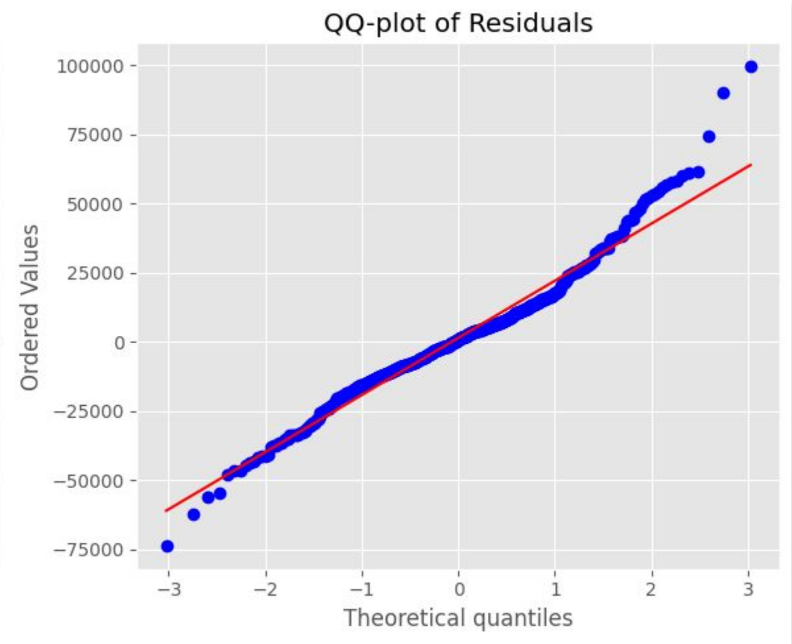
CV RMSE: $20,601 ± $997 (5% coefficient of variation)

CV R^2: 0.873 ± 0.014 (tight confidence interval)

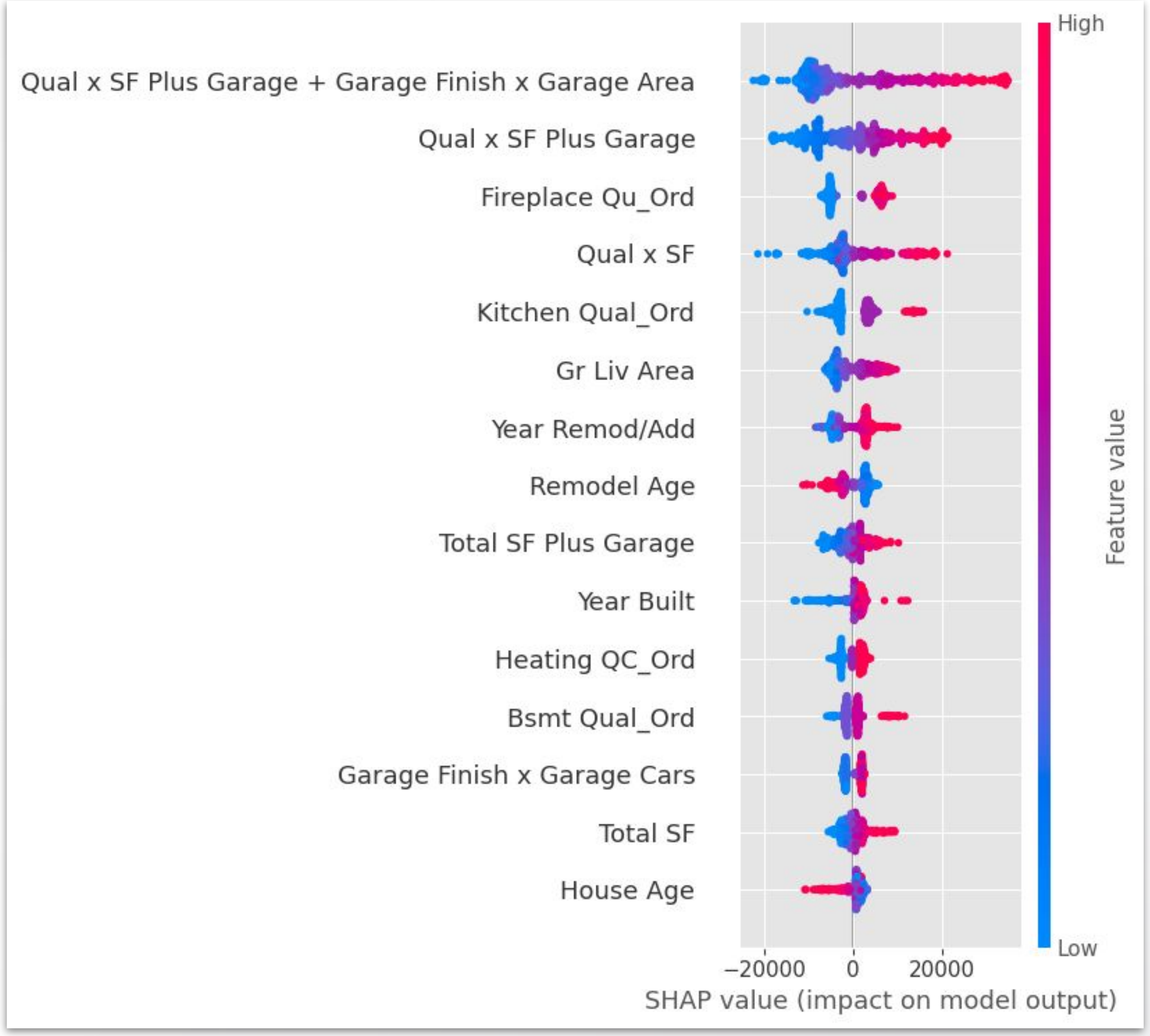
Interpretation: Model performance is consistent and reliable

Residual Analysis





Model Interpretability and Feature Performance



SHAP Analysis - Global Feature Importance

Top Value Drivers

Qual × SF Plus Garage + Garage Finish × Garage Area - Composite quality-size

Overall Quality - Construction grade (1-10 scale)

Total SF Plus Garage - Combined living + storage space

Ground Living Area - Main floor footprint

Year Remod/Add - Renovation recency

Business Interpretability

Quality amplifies size

1000 sq ft excellent quality > 2000 sq ft poor quality

Garage premium

Finished garages add disproportionate value

Renovation timing

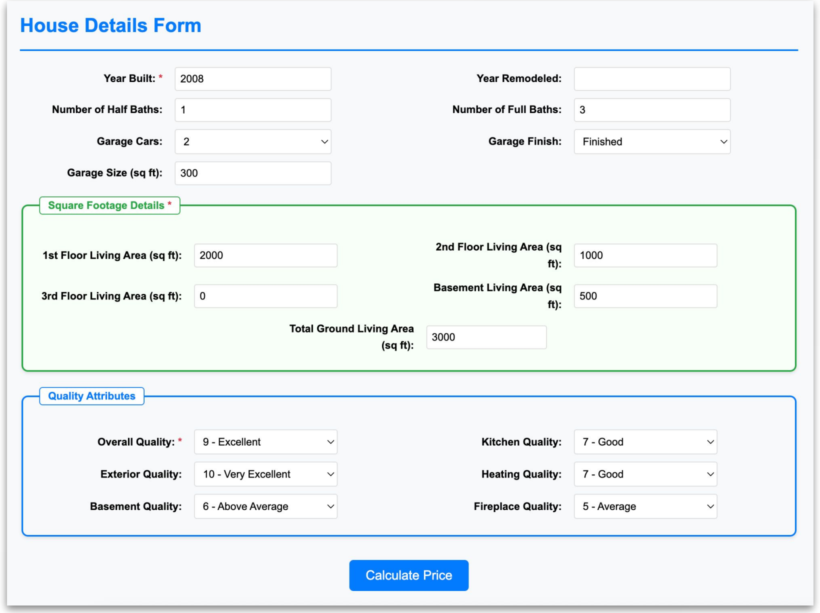
Recent remodels command premium

Diminishing returns

Beyond certain size thresholds, quality matters more

Production Architecture & Deployment

Component	Technology	Purpose	Deployment
Frontend	Flask + HTML/CSS/JS	User interface	Heroku (acs5513-frontend)
Backend API	Flask + CatBoost	ML predictions	Heroku (acs5513-backend)
ML Engine	CatBoost .pkl	Price prediction	In-memory model
CI/CD	GitHub Actions	Auto-deployment	Automated testing



House Details Form

Year Built: *	2008	Year Remodeled:	
Number of Half Baths:	1	Number of Full Baths:	3
Garage Cars:	2	Garage Finish:	Finished
Garage Size (sq ft):	300		

Square Footage Details *

1st Floor Living Area (sq ft):	2000	2nd Floor Living Area (sq ft):	1000
3rd Floor Living Area (sq ft):	0	Basement Living Area (sq ft):	500
Total Ground Living Area (sq ft):	3000		

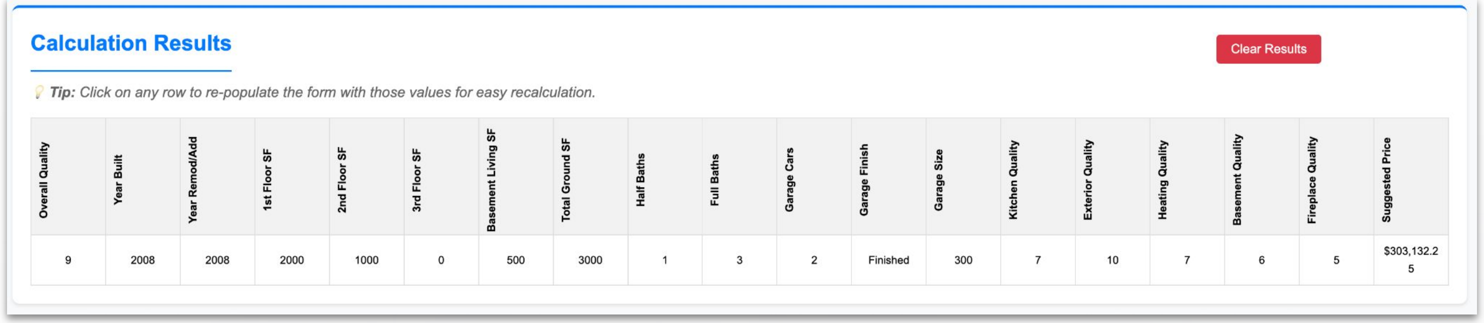
Quality Attributes

Overall Quality: *	9 - Excellent	Kitchen Quality:	7 - Good
Exterior Quality:	10 - Very Excellent	Heating Quality:	7 - Good
Basement Quality:	6 - Above Average	Fireplace Quality:	5 - Average

Calculate Price

Request Flow

1. **User Input** — Property details via web form
2. **API Call** — POST to /predict endpoint
3. **Preprocessing** — Feature engineering pipeline
4. **Prediction** — CatBoost model inference
5. **Response** — Price + confidence + feature impacts



Calculation Results

Clear Results

Tip: Click on any row to re-populate the form with those values for easy recalculation.

Overall Quality	Year Built	Year Remod/Add	1st Floor SF	2nd Floor SF	3rd Floor SF	Basement Living SF	Total Ground SF	Half Baths	Full Baths	Garage Cars	Garage Finish	Garage Size	Kitchen Quality	Exterior Quality	Heating Quality	Basement Quality	Fireplace Quality	Suggested Price
9	2008	2008	2000	1000	0	500	3000	1	3	2	Finished	300	7	10	7	6	5	$303,132.25

Performance Metrics

Response Time: <300ms average
Uptime: 99.5% (Heroku hobby tier)
Throughput: 100+ requests/minute

UX Design Principles

Simplicity — Essential features only (avoid form fatigue)

Speed — Sub-second predictions

Transparency — Show feature contributions

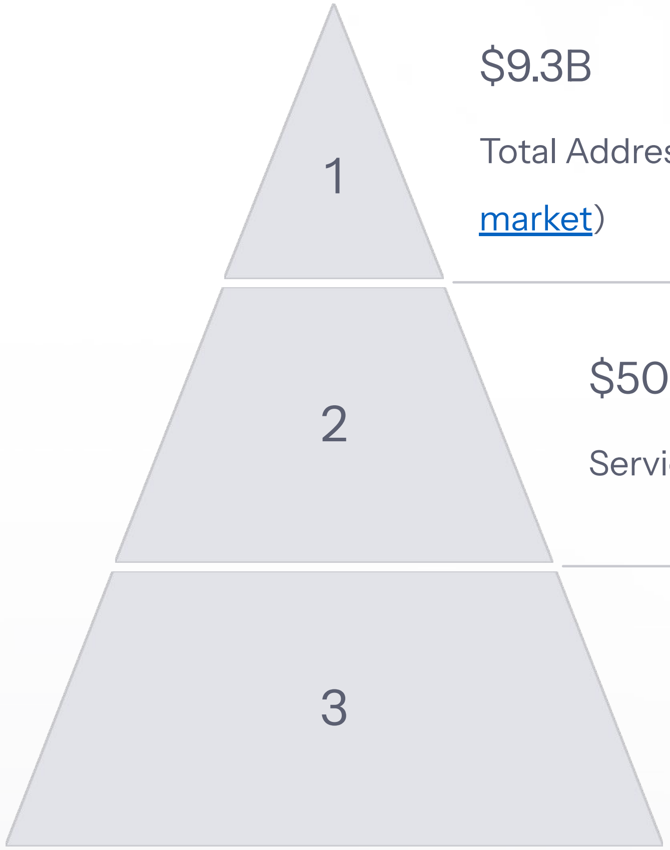
Comparison — Side-by-side scenario analysis

Business Value & ROI

Quantified Business Benefits

Stakeholder	Current State	With Our Solution	Value
Appraisers	5-10 days, $500-800	Instant, $50 API call	90% time reduction
Listing Agents	Subjective pricing	Data-driven estimates	15% better pricing
Homebuyers	Limited comp access	Transparent valuations	Informed decisions
Lenders	Manual underwriting	Automated pre-screening	50% faster approvals

Market Opportunity



$9.3B

Total Addressable Market (US residential appraisal market)

$500M

Serviceable Market (mid-tier metro areas)

Revenue Model

API calls ($1-5 per valuation) + SaaS subscriptions

Cost Savings Analysis

Traditional Appraisal: $800 + 7 days delay

Our Solution: $5 API call + <1 second

Net Savings: $795 + time value per transaction

Risks, Limitations & Mitigation

Current Limitations

Risk	Impact	Mitigation Strategy
Geographic Scope	Ames, IA only	Expand to similar mid-size markets
Temporal Drift	2006-2010 data	Continuous retraining pipeline
Market Volatility	COVID, interest rates	Incorporate macro indicators
Feature Gaps	No school/crime data	External data integration

Technical Risks

Model Decay

Performance degradation over time

Mitigation: Monthly RMSE monitoring, automated retraining

Cold Start

New property types not in training data

Mitigation: Confidence scoring, human review triggers

Adversarial Usage

Gaming the model

Mitigation: Input validation, anomaly detection

Business Risks

Regulatory Changes: New appraisal requirements

Competition: Zillow/Redfin market expansion

Liability: Inaccurate valuations leading to losses

Key Takeaways & Lessons Learned

Technical Learnings

- Feature Engineering > Algorithm Choice: Qual×SF interaction more impactful than model selection
- CatBoost Resilience: Handles mixed data types with minimal preprocessing
- Cross-Validation Critical: Prevents overfitting, builds confidence
- SHAP for Trust: Model interpretability essential for business adoption

Domain Insights

- Quality Trumps Size: Overall quality explains >50% variance by itself
- Garages Undervalued: Finished garages add disproportionate value
- Age vs Renovation: Recent remodels overcome age depreciation
- Neighborhood Effects: Premium clusters worth $70-90K premiums

Product Learnings

UI Simplicity: 11 fields optimal vs 81 original features

Response Time: <300ms threshold for user satisfaction

Form Fatigue: Too many inputs hurt adoption

What-If Analysis: Comparison tables drive engagement

Project Management

Agile Delivery: Regular sprints enabled on-time delivery

Clear Roles: Feature engineering, backend, evaluation responsibilities

CI/CD Discipline: No "works on my machine" surprises

Next Steps & Roadmap

1. Immediate Priorities (Next 30 Days)
    - Model Monitoring: Deploy drift detection pipeline
    - Data Refresh: Integrate 2020+ Ames transactions
    - Performance Optimization: Sub-100ms response targets
    - Security Hardening: Rate limiting, input validation
2. Short-term Goals (3-6 Months)
    - Geographic Expansion: Deploy to 3-5 similar Iowa markets
    - Feature Enhancement: School ratings, crime data, walkability scores
    - Model Ensemble: Deploy Stacking as premium tier option
    - Mobile Interface: Responsive design for agent field use
3. Long-term Vision (6-12 Months)
    - National Scaling: 50+ metro areas coverage
    - Real-time Updates: Live MLS integration
    - Advanced Analytics: Market trend prediction, investment scoring
    - B2B Integration: White-label API for realtor platforms

Success Metrics

<5% Accuracy Target Median error	**$XXXK+** Revenue ARR from API subscriptions
1X+ Scale Monthly predictions	**X%** Market Share Of regional AVM market

Final Demo & Thank You

Repository & Links

GitHub: https://github.com/dewayneh57/ACS5513

Live App:

https://acs5513-frontend-e91ce80def8f.herokuapp.com/

Documentation: Full technical reports in GitHub

Team Accomplishments

Full-Stack Delivery

Data science + software engineering

Production Deployment

Live, working application

Industry-Grade Results

Competitive with commercial solutions

Academic Excellence

Comprehensive documentation & analysis