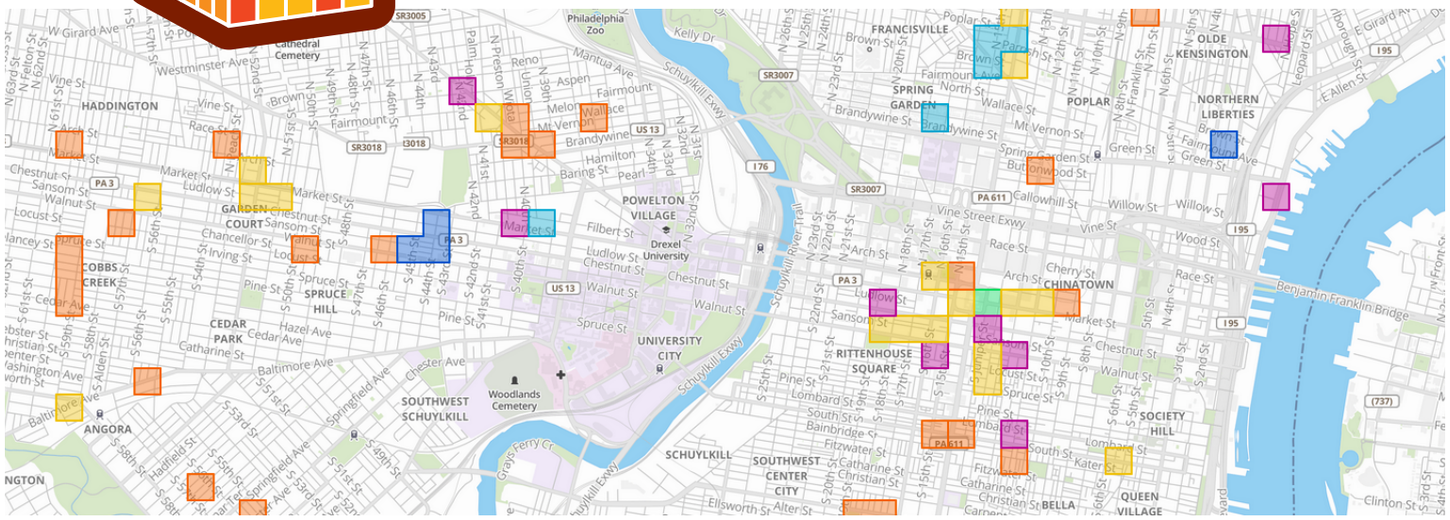
HunchLab



HunchLab: Under the Hood

HUNCHLAB: UNDER THE HOOD

Version: 1.1.0

For more information: http://www.hunchlab.com

FOREWORD

Current trends reflect an interest in predictive policing, increasing adoption of cloud services, pervasive location information, and the proliferation of mobile devices. These trends are also being driven by an extremely constrained budget environment in most law enforcement agencies. We believe these factors will combine to cause the law enforcement community to undergo a transformational change in the use of technology, similar in scope to the advent of GIS-based crime mapping in the 1990's. The new HunchLab anticipates these advances in an unprecedented manner. We believe that HunchLab 2.0 truly represents the future of predictive policing.

We invite you to join us to help craft a tool of exceptional usefulness for police departments worldwide to improve public safety.

It’s the fourth Tuesday in January and school is in session. There were 3 burglaries and 2 robberies yesterday. Six bars, three take-out stores, and a school are in the neighborhood. The forecast is 17° with cloudy skies.

Where do you focus your 2 patrol vehicles?

CHAPTER 1

Overview

What if a crime analysis system helped you to focus on what matters each day and helped you to apply evidence-based tactics to improve public safety? What if a software vendor rethought how to design policing software from the ground-up?

HunchLab began as a project for the Philadelphia Police Department and the Office of the U.S. Attorney. They asked Azavea to develop a "Crime Spike Detector" – a geographic change detection system that could sift through millions of records each day and identify statistically significant spikes in clusters of crime events in the City of Philadelphia. Based on the success of the Philadelphia prototype, Azavea was able to win a Small Business Innovation Research (SBIR) grant from the National Science Foundation to support the development of a next generation Crime Spike Detector software tool – HunchLab – that could be used by police departments around the country.

Development of this first version of HunchLab proceeded from 2008 through 2011 and included early versions of forecasting capabilities based on techniques published by various academic researchers including near repeat pattern analysis and crime load forecasting based upon seasonality. Since this first version of the software, Azavea began working with Drs. Jerry Ratcliffe and Ralph Taylor at Temple University to model long-term crime trends based upon neighborhood demographic indicators. With Drs. Joel Caplan and Les Kennedy at Rutgers University, Azavea worked to automate their Risk Terrain Modeling approach to crime forecasting which explains why crimes emerge at specific locations based upon the nature of those locations (proximities to bars, bus stops, etc.)

These endeavors inspired Azavea to ask two questions:

- Could we create a system that uses many crime theories to generate a unified prediction of crime?
- How could such a unified picture of risk transform the way we think about crime analysis and predictive policing software?

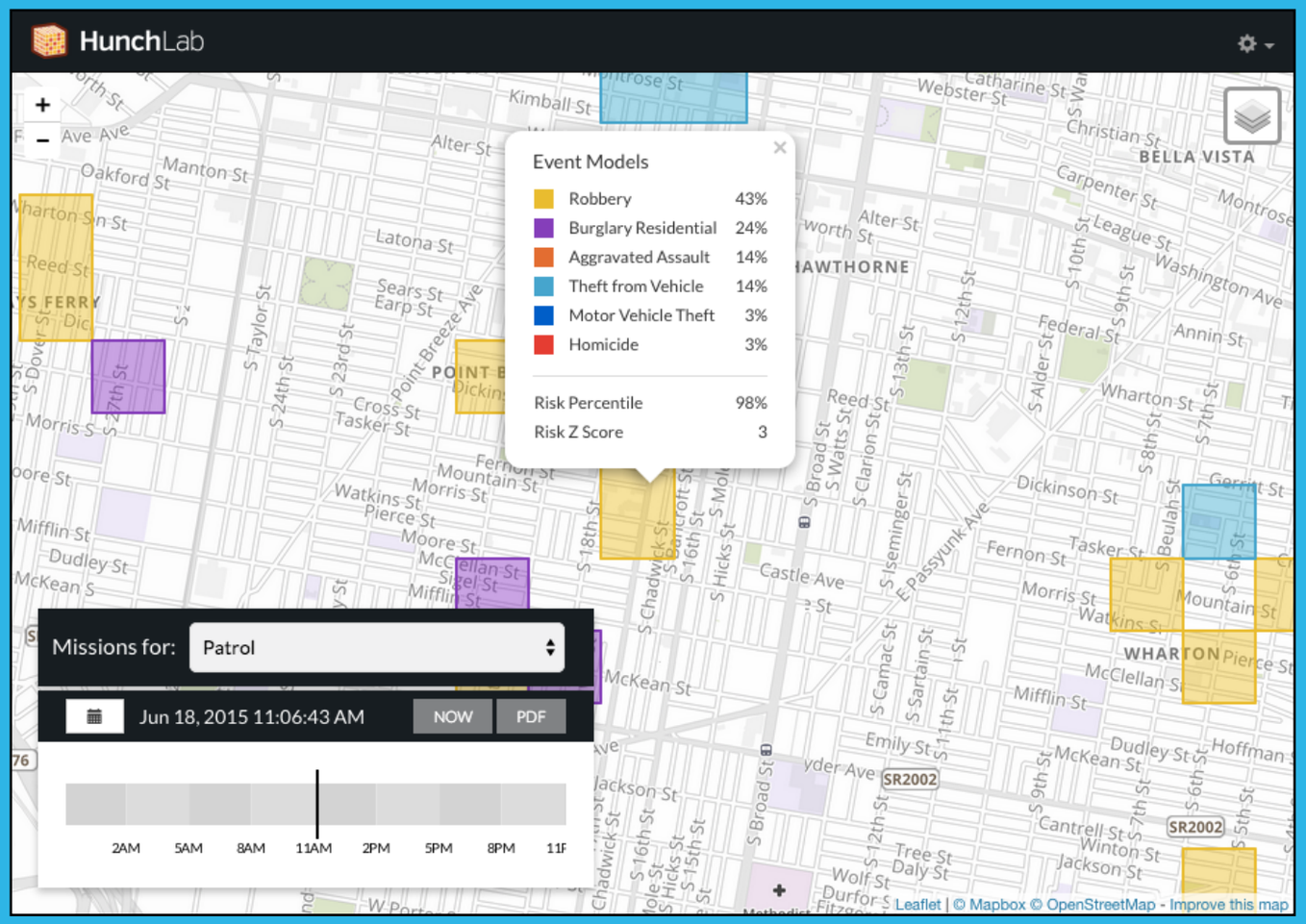
HunchLab 2.0 represents our answer to these questions by providing functionality for command staff, analysts, and officers.

Command Officers Analysts

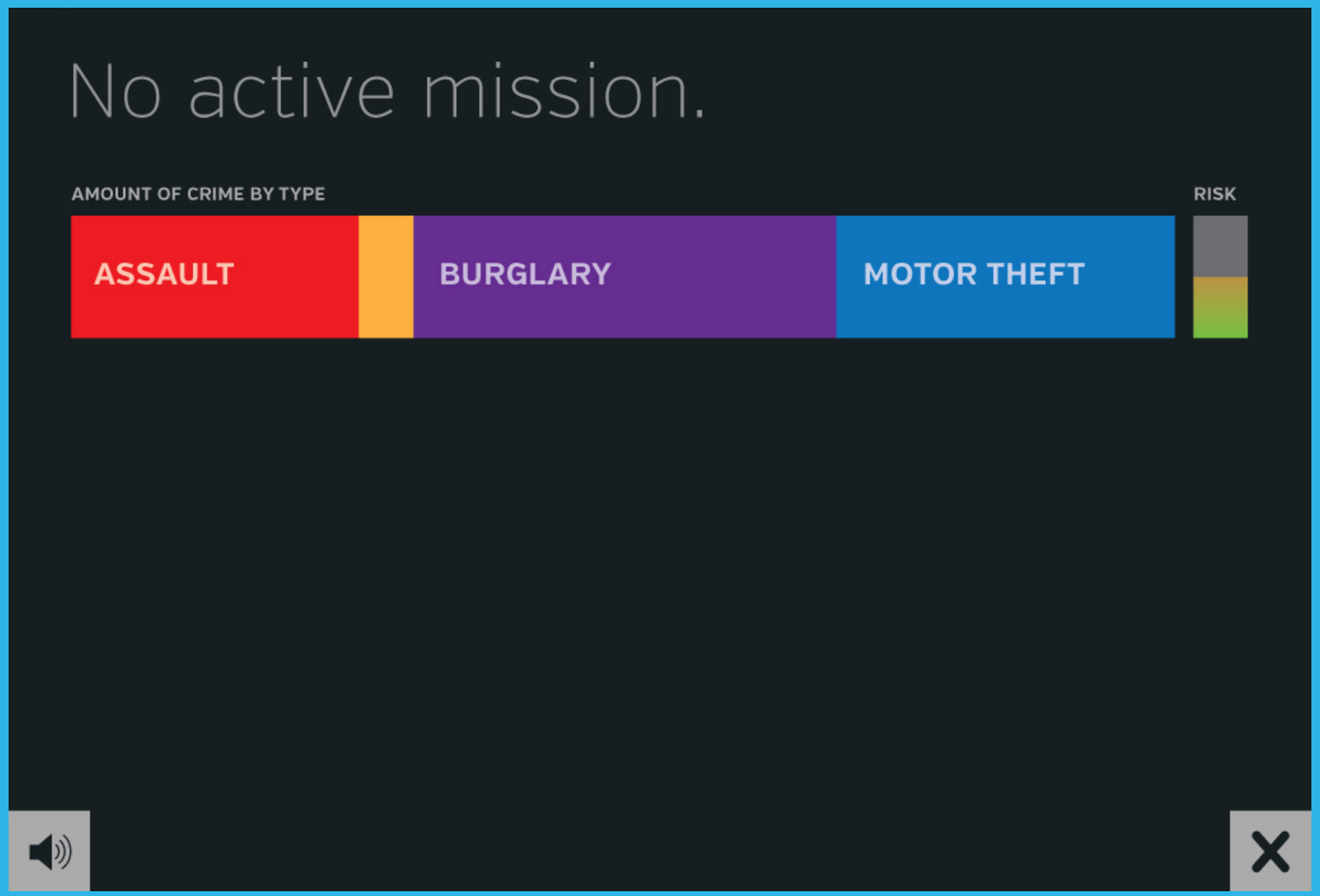
Crime Models

	Label	Severity Weight	Patrol Efficacy	Patrol Weight	Relative Weight	
	Homicide	8,649,216	1%	86,492.2	53.9	
	Aggravated Assault	87,238	5%	4,361.9	2.7	
	Robbery	67,277	20%	13,455.4	8.4	
	Motor Vehicle Theft	9,079	50%	4,539.5	2.8	
	Theft from Vehicle	2,139	75%	1,604.3	1.0	
	Burglary Residential	13,096	25%	3,274.0	2.0	
	Gun-related Crimes	100,000	15%	15,000.0	9.4	

Command staff set crime priorities used by the system to generate predictive mission areas. Mission areas are designed to maximize the benefit of resource deployments across this basket of weighted crime types.



Mission areas represent the highest risk locations during the day's shift and reflect the quantity of resources available for that day's deployment. Color is used to highlight the crime focus for each mission area.



Officers receive information about forecasted risk at any location within the jurisdiction. In this example, the current location's primary risk is burglaries followed by motor vehicle thefts.

Predictive Analysis

We have spent the last few years determining how to incorporate multiple crime theories into one forecast. For example, we can incorporate concepts such as: temporal patterns (day of week, seasonality); weather; risk terrain modeling (locations of bars, bus stops, etc.); socioeconomic indicators; historic crime levels; and near repeat patterns. The system automatically learns what is important for each crime type and provides recommendations of where to focus the resources that you have available. If you don't have particular datasets (such as bars or bus stops), the system simply adapts to use the data available in a given jurisdiction.

Deployment Planning

Armed with this new predictive capability we also designed a system to generate mission recommendations by considering the number of available staff and vehicle resources. After predicting crime expectations across the jurisdiction, our solution calculates the forecasted crime level per unit of patrol effort for each area, sorts these areas from highest to lowest relative risk, and selects the mission areas that can be patrolled by the available resources. This process maximizes the impact of patrols and ensures that the right quantity of mission areas is crafted.

Leveraging Analysts

While predictive policing can automate the selection of focus areas for deployments, they do not replace the work of skilled analysts. In HunchLab 2.0, we enable analysts to support evidence-based practices in two ways. First, working with command staff, analysts design tactical responses to specific crime types and location types; randomly patrolling a focus area is not always the best course of action. By publishing tactics, analysts help officers to make effective use of their time. Second, analysts publish spatial notes to the field. Perhaps an offender was recently released from prison or perhaps the analyst has a suspect in mind for a recent burglary series. Spatial notes capture this information and disseminate it to the field when it is most relevant.

Mobility

Once generated, these mission areas can be disseminated via both low and high tech means. The simplest dissemination technique is a printable PDF report outlining mission areas for the shift. For departments with mobile broadband and GPS-enabled MDTs, smartphones, or tablets we provide a location-based service called Sidekick. Sidekick provides officers with crime predictions about their current location, notifies them of mission objectives, and supports measurement of the dosage of field tactics to address crime problems.

CHAPTER 2

Predictive Analysis

Creating accurate predictive models is at the core of HunchLab 2.0. By forecasting crime risk, we assist a police department to more efficiently allocate their resources, which helps them to make a bigger impact. While predictive analysis may seem magical, at the end of the day it boils down to a simple concept: learn from the past to better anticipate the future.

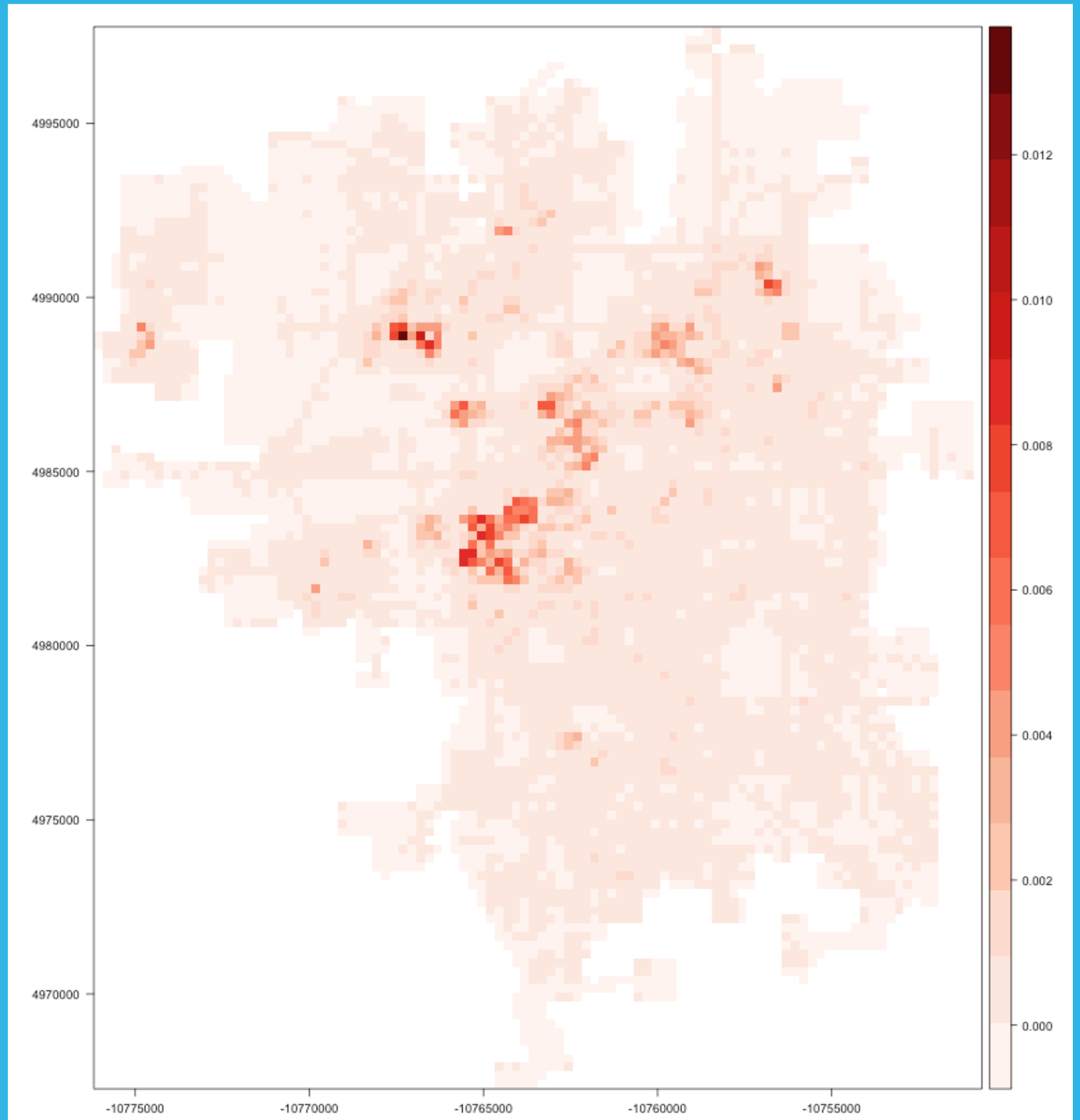
Geographic Predictive Analysis

At the core of HunchLab 2.0 is a new crime forecasting engine. These predictions power the new predictive missions capability, which enables departments to proactively generate the appropriate quantity of mission areas based upon the organizational and societal importance of various types of crime. The forecasting engine uses ensemble machine learning approaches that can incorporate the following crime patterns into a single prediction of criminal risk:

- Baseline crime levels
  - Similar to traditional hotspot maps
- Near repeat patterns
  - Event recency (contagion)
- Risk Terrain Modeling
  - Proximity and density of geographic features (points, lines, and polygons)
- Routine activity theory
  - Offender: Proximity and concentration of known offenders
  - Guardianship: Police presence (historic AVL / GPS data)
  - Targets: Measures of exposure such as population, parcels, or automobiles
- Collective Efficacy
  - Socioeconomic indicators, heterogeneity, etc.
- Temporal cycles
  - Seasonality, time of month, day of week, time of day, etc.
- Recurring temporal events
  - Holidays, sporting events, etc.
- Weather
  - Temperature, precipitation, etc.

Our belief is that the use of non-crime data sets as variables within a crime prediction system is important, because variables based solely upon crime data become skewed as predictions are used operationally. For instance, as crimes are prevented in mission areas due to police response, the only variables identifying areas as high risk are skewed in other systems. By including other data sets, our system is more robust against this issue.

Our approach to mission recommendation is priority and resource-aware. After predicting individual crime expectations across the jurisdiction, our solution combines the individual predictions based upon department priorities. For instance, assaults may be more important than burglaries. The system then calculates the relative crime level per unit of patrol effort for each area, sorts these areas from highest to lowest relative risk, and selects the amount of areas that are able to be patrolled by the resources available. This process maximizes the impact of patrols and ensures that the right quantity of mission areas is crafted.



This is an example visualization of the predictions that HunchLab uses to generate mission areas; each cell has a predicted crime expectation.

CHAPTER 3

Frequently Asked Questions

We believe that collaboration and transparency will move policing forward faster than patent-pending algorithms and black-box software products. We've assembled answers to questions we frequently receive about HunchLab to describe its origins and explain how it works under the hood.

When was HunchLab first introduced?

The HunchLab project began in 2005 as a prototype to detect localized spikes in crime activity. Azavea then secured research & development funding from the National Science Foundation to expand this prototype into a commercial product. Development of this first version of HunchLab proceeded from 2008 through 2011 and included early versions of basic forecasting capabilities. In 2013, we developed a new statistical approach to forecasting crime that could include multiple data sets and crime theories in one picture of risk. This new statistical approach was combined with a new design to form the HunchLab 2.0 application and previewed publicly in September 2013. The first HunchLab 2.0 pilot clients were deployed at the start of 2014.

Many predictive policing technologies seem to have originated from techniques unrelated to crime. How did HunchLab originate?

In Azavea's various civic software projects, we have found that blending knowledge from two distinct fields often results in innovation. For instance, in HunchLab 1.0, we took a statistical method often used in bio-surveillance to detect disease outbreaks and applied it to crime data to detect spikes in activity. From a risk forecasting perspective, our analytic techniques within HunchLab 1.0 were based upon published academic work that each looked at an individual aspect of crime patterns.

Azavea began receiving feedback about ways to enhance these features. For instance, our load forecasting feature in HunchLab 1.0 forecasts aggregate crime levels based upon season, day of week, and the overall trend. When users saw this feature, they immediately asked for weather and special events to be included in the forecast. This type of feedback led us to begin work on HunchLab 2.0.

As we developed HunchLab 2.0, we started with a desire to incorporate different data sets (each representing a different crime theory perspective) into a unified forecast. We then evaluated a number of machine learning and statistical models to solve this problem and evaluated how they met our design objectives. Some of the most innovative approaches to predicting outcomes are, indeed, being developed for use in other fields including e-commerce and advertising. There is not a specific inspiration for the statistical approach that we are using (detailed in a later answer). The modeling approach we are using is used in many domains and is often found to be the most accurate predictive algorithm.

What criminological theories have shaped HunchLab 2.0?

As the previous answer mentioned, in developing HunchLab 2.0, we had a design objective to be able to incorporate various criminological theories into one unified model. These theories include concepts such as the near repeat phenomenon discussed by a number

of academic researchers over the years and more recently re-branded under the "self-exciting point process" moniker.

We also aimed to include concepts such as the Risk Terrain Modeling research being published by Rutgers University, which describes crime locations through correlated geographic features such as bars, schools, and transit stops.

HunchLab 2.0 includes each crime theory by deriving individual sets of variables that represent the concepts within each theory. For instance, we may represent Risk Terrain Modeling by measuring the distance to the nearest bar and the density of bars at each raster cell. We may represent the near repeat pattern concept by measuring the amount of time since the most recent crime occurred in each raster cell. These sets of variables are then passed into the modeling process, which determines the useful theories for a given crime type. The system also determines how the theories interact. For instance, if the near repeat pattern effect is stronger in areas with lots of historic crimes, the system can use that information to enhance the forecast. Or, for example, if assaults frequently happen on Friday evenings near bars, the system can model that effect.

Analysts are able to select which data sets are used to model a particular type of crime. For instance, you could decide to give the assault modeling process access to all geographic layers, in which case the system will include the ones with which it finds correlations. Alternatively, you could pre-screen the geographic layers and exclude a particular layer because you do not feel there is a theoretical basis for its use with assaults.

What modeling methods does HunchLab use to generate forecasts of crime?

In some ways, the model building process in HunchLab mimics the thought process of an experienced analyst. For instance, consider asking an analyst to decide where to place patrol resources for a given upcoming time period. She may start by looking at where crimes have occurred in concentration previously and delineate hotspots of activity. Based on her past experience, she may know that during this particular time period, schools dismiss their students, which increases petty crimes around the schools in the neighborhood. She builds up many such layers of knowledge and balances these various concerns to form a plan. After the time period concludes, she may go back and look at where activities occurred to see if she can determine additional insights into the crime patterns to include in future plans. HunchLab incorporates machine learning concepts to help the software "think" like a crime analyst by imitating years of experience drawn from a police department's own data.

The concept of machine learning is to teach a computer to accomplish a particular task. In this case, we want to teach the computer to determine how likely a particular crime type is to occur at various locations for a given time period. We start this process in HunchLab by forming a set of training examples using the past several years of crime data. Each training example contains the theoretically derived variables we explained above, as well as the outcome (how many crime events occurred). For an entire municipality this training set will often include many millions of example observations. We can then start building the model.

The primary model HunchLab currently uses is a stochastic gradient boosting machine (GBM) comprised of decision trees trained using the AdaBoost loss function. This model is built to forecast whether a crime event will occur or not in a given space-time raster cell (a binary outcome). The general way this model works is as follows:

- Begin by selecting a random portion of the training examples.
- Build a decision tree that separates examples of where crimes occurred from ones that didn't based upon the variables.
  - For instance, the first decision within the tree might be interpreted as: "if no event happened in the last year in this location, it is very unlikely for a crime to occur today". The decision tree then splits the examples into two sets: (1) where a crime occurred during the past year and (2) where no crimes occurred during the past year.
  - Within each set, the process repeats. For example, the next decision for the set of locations with crimes in the past year might be interpreted as: "if an event happened in the last week, it is more likely for one to occur today". This set of examples would again be split based upon this decision rule.
  - This process continues to build out a decision tree that describes why crimes occur where they do.
- The decision tree is then used to make predictions of how likely crimes are for each observation in the entire training data set.
- This completes one training iteration within the boosting machine.
- The modeling process then begins again.
  - We start by selecting another random portion of the training examples. This random sampling process is why the model is stochastic.
  - In this next iteration, we build another decision tree (in the same manner as above). This time, however, we build the tree to predict the errors from applying the first decision tree model to this new sample of observations. In this way we are attempting to correct our mistakes. This concept is called boosting.
  - We then use these two trees to make predictions across the entire data set.
  - As we conduct this process, we can keep track of how many training iterations within the machine have made incorrect predictions for each training example. We increase the importance (via weights) of observations that we continue to get wrong and decrease the importance of observations that we continue to get correct. This process is called adaptive boosting (AdaBoost). When we build the next decision tree, we tell it to focus on the observations that we continue to get wrong via these weights.
- Training iterations continue several hundred times. The resulting model represents tens of thousands of decision rules of why crimes occur where they do.
- We conduct this entire process several times, each time holding back a portion of the example data. We can then use each of these models to make predictions for this held-out set of data to

see how accurate the model is when we apply different quantities of training iterations from the model. For instance, if the models have 100 training iterations, we may find that the most accurate predictions come from only using the first 53 iterations. This process is called cross-validation and prevents our models from over-fitting the training data.

- Finally, we begin the entire process again using the whole data set to build a model with the correct number of training iterations. In this example, we would use 53 iterations.

As you can see, this modeling process mimics some activities that an analyst would go through in making decisions of where to focus resources. The predictions from this model are whether one or more crimes will occur or not. We then need to translate these probabilities into expected counts. We do this by calibrating our predictions using a generalized additive model that assumes a Poisson distribution. This regression model both translates the outputs of our model to expectations and calibrates the predictions. For example, the above model might slightly over-predict crimes on Tuesdays. This calibration step would lower the predictions for Tuesdays to center them on the training data. The process of using one model's outputs as another model's inputs is called model stacking.

These models are then saved and used to generate predictions. The predictions are calibrated count expectations for each raster cell for a given period of time. You may picture predicted counts to be numbers such as 0, 1, 2, or 3. In actuality, the predictions are real numbers that are often fractions such as 0.000001, 0.02142, or 0.12482. This represents the fact that the nature of crime is such that no software solution can say that a crime is going to happen at this specific corner at this precise time. For a small raster cell and time period, it is almost always more likely that no crime will occur. What is important is that we can use these predictions to measure the relative risk of events between locations, time periods, and different crime types, so that we focus on the most likely types of events at the most likely locations and times.

Aoristic Times

Often the exact time that a crime event occurred is unknown. A prime example of this phenomenon is residential burglary, where the homeowner discovers the burglary when they arrive home from work. We handle this phenomenon by using aoristic time ranges within our model. For each crime event imported into HunchLab, you can provide a time span during which the event occurred (sometime between 9:00 AM and 5:00 PM, perhaps). If the time of the event is certain, these start and end times are simply set to the same time.

When HunchLab builds the training data set that is used in the modeling process described above, we use these time ranges to determine the possible outcome scenarios for the observation. For instance, assume we are creating a training example for a particular raster cell from 9:00 AM to 10:00 AM on January 16, 2014. There is one crime that may have occurred during this period. The event happened between 9:00 AM and 11:00 AM. We therefore have two scenarios from 9:00 AM to 10:00 AM: (1) 0 events occurred and (2) 1 event occurred. We assume that the event is uniformly likely to have occurred during the aoristic time frame (2 hours in length), so each of these scenarios is equally likely (1 hour / 2 hours = 50% probability). Both scenarios are placed into the training example

set for use in modeling with a weight of 0.50 for each scenario. As the modeling process selects random portions of examples for training during each training iteration, it will sometimes include the first scenario and sometimes the second one. This approach nicely represents the relative uncertainty of the exact event time.

Model Variations

There are several parameters that define the exact functionality of the model building process described above. HunchLab can also use varying amounts of historic data in building the model in order to balance the desire to have more examples with the desire to use recent examples. We adjust some of these various parameters to generate a few variations of our modeling process. The system automatically scores each variation on a held-out data set (the most recent 28 days of crimes, for instance) and then selects the best performing variation for use in production.

Measuring Accuracy

The process that selects among the model variations also allows us to measure the accuracy of our predictions. We measure not only among the model variations but also in comparison to several baseline models that reflect standard approaches to deploying resources (for example kernel density maps generated over varying time periods). Azavea can run crime data sets for potential clients through our modeling process and provide measurements of the accuracy to them. Accuracy varies based upon data quantity (more is better), quality (cleaner is better), and the nature of the crime type.

What data is used to generate crime forecasts? Does HunchLab solely use police data or does it analyze data from other sources as well?

HunchLab 2.0 can use both police data and other data sets to generate crime forecasts. The only required data set is the crime event data itself (the outcome we are forecasting). We test our statistical models with just this data set being available to ensure that we can produce accurate forecasts if this is the only data that a specific police department has available.

On the other hand, if additional data sets are available, we wanted to be able to use them to generate more accurate forecasts. Azavea believes that systems that can only utilize crime data suffer from several problems. For instance, historic locations of certain types of crime are not very useful in forecasting future events (i.e. rape). Second, once a police department begins acting upon the predictions from a statistical model, they alter the outcomes (crime events are prevented). This feedback loop then skews the only variables being used to predict risky locations. By including other data sets in HunchLab's modeling process, we can diversify our picture of criminal risk and mitigate the effect of these issues.

Some data sets we manage on behalf of clients. For instance, we automate the use of national holidays and weather within our forecasts. Other data sets might be procured by the police department from another government agency (example: business permits for locations of restaurants or liquor establishments) or developed internally (example: gang member residences).

These additional data sets fall into two main categories:

1. geographic data sets containing points, lines, and polygons
2. temporal data sets such as school schedules, social events, or weather data.

How do HunchLab's forecasts support decision making?

Our current functionality in HunchLab 2.0 focuses on forecasts of specific crimes in specific geographic locations. For instance, an analyst may configure a crime model within HunchLab to forecast residential burglaries. HunchLab would then automatically produce predictions (count expectations) of residential burglaries in each raster cell (cell size is configurable, but each cell is often 100-250m in size) for the upcoming hours. The temporal granularity can range from a one-hour block of time to one shift.

A particular police department would configure several crime models within HunchLab based upon the manner in which they organize themselves. For instance, you would likely create a model for each major crime type as well as additional focus crimes that you are looking to address. The system generates separate predictions for each crime model that you configure. These separate predictions are then combined to create target areas based upon the crime weights set by your department. These weights allow you to prioritize crime types based upon the impact each crime type has on your community.

Azavea is also in the research and development phases for a 2nd module in HunchLab 2.0 that would model the general ex-offender population. This model would predict the likelihood of each offender committing another crime and would be useful for proactive outreach (such as parole and probation activities), as well as prioritizing suspects for investigation in connection with new crime events.

Where is HunchLab currently deployed?

Prior versions of HunchLab have been deployed across the country. It has been deployed for several years in the City of Tacoma and Pierce County, Washington, for the City of Philadelphia Police Department and for the Northwest Ohio Regional Information System (NORIS). A demo instance has also been implemented for Lincoln, Nebraska.

HunchLab 2.0 is being deployed in several places including Philadelphia, PA, New Castle County, Delaware, and Lincoln, Nebraska. We are piloting the software with additional municipalities, including a large European city, but are often under non-disclosure agreements during trial periods. If your agency is interested in piloting HunchLab and helping us to expand on our capabilities, please contact us.

Is HunchLab based on proven research? Has it been documented in the literature?

Since HunchLab 2.0 is new, Azavea does not yet have published evaluation documents. The theoretical concepts used within HunchLab 2.0 are not new, however, and have been studied and documented by many academics. At the end of this section, we have provided links to some research papers that we and others in the field have written on Risk Terrain Modeling, near repeat forecasting, and other methodologies that have been operationalized in HunchLab. Individually, these methodologies have demonstrated their effectiveness in a number of police departments in North America and Western Europe, but research is still ongoing.

For HunchLab 2.0, Azavea views effectiveness as two main components. The first component deals with the accuracy of the predictions themselves. This component we are measuring for various crime types in an ongoing fashion and can make claims about. The second component deals with the effectiveness of a predictive policing tool in terms of crime reductions. It is tricky to use this measure in selecting a solution because it is so dependent on how the tool is used by an agency. Software, after all, doesn't prevent crime. For instance, we could have the same tool at two law enforcement agencies. The culture of the first agency is very data driven, and they adopt the tool to great success. The command structure at the second agency doesn't value the tool and therefore it has no impact.

Related Publications

Here are some publications of interest on the crime theories used in HunchLab:

Near repeat pattern analysis

Haberman, CP & Ratcliffe, JH (2012) The predictive policing challenges of near repeat armed street robberies, Policing: A Journal of Policy and Practice.

Ratcliffe, JH & Rengert, GF (2008) Near repeat patterns in Philadelphia shootings, Security Journal. Volume 21, issue 1-2: 58-76.

Risk Terrain Modeling

Heffner, J. (2013). Statistics of the RTMDx Utility. In J. Caplan, L. Kennedy, and E. Piza, Risk Terrain Modeling Diagnostics Utility User Manual (Version 1.0). Newark, NJ: Rutgers Center on Public Security.

Additional resources (publications, software, manuals)

Seasonality

Wilpen Gorr , Andreas Olligschlaeger , Yvonne Thompson (2003) Short-term Forecasting of Crime, International Journal of Forecasting 19

CHAPTER 4

Security

Delivered as a multi-tenant software-as-a-service (SaaS) solution, all clients benefit from the robust security design of the HunchLab 2.0 application, which also simplifies application roll-out.

Azavea has a long history of handling sensitive law enforcement data sets. The new version of HunchLab is delivered as a secure cloud-based subscription service. As we designed this new version, we focused on incorporating security best practices into our development process. While most deployments of HunchLab contain local department data sets that do not technically require compliance with the FBI's Criminal Justice Information Systems (CJIS) guidelines, we are using the CJIS requirements and recommendations to guide our decision making process and system architecture. While we are not certifying full CJIS compliance in this current pilot phase of HunchLab 2.0, here are some of the security features and policies available within the new HunchLab.

Data Use & Security Agreement

By default, Azavea agrees to solely use the law enforcement data to provide the agreed upon HunchLab service to the department. This includes using the data for system testing, refinement, and live operations. Separately, Azavea may seek permission to use the data for research purposes that further the product and crime analysis in general. At no time will Azavea hold any claims to the data nor will Azavea use the data for other commercial purposes. Upon written request, Azavea will purge a customer's law enforcement data from its systems.

Azavea will gladly sign a CJIS Security Addendum as specified in CJIS v5.1 section 5.1.1.5.

Security Awareness Training

Azavea hires technical staff with an eye toward building reliable and secure web applications. Part of the Azavea onboarding process is acknowledgement of company security practices as well as signing a separate agreement regarding confidentiality of client data. Additionally, staff with access to the HunchLab system undergo biennial training on best practices when dealing with criminal justice information, as outlined in CJIS v5.1 section 5.2.

Reliability & Security Incident Management

The HunchLab service is designed to be resilient to failure with redundancy built into the system architecture. Additionally, Azavea has implemented automatic monitoring of system uptime and incident alerts to provide timely resolution of system issues. In the event of a security breach, Azavea will proactively notify the law enforcement agency of the breach in a timely manner as specified in CJIS v5.1 section 5.3.2.

System Auditing

The HunchLab system keeps a running system log of activity by users, including log-on attempts and information retrieval. These records are retained for at least 365 days. The auditing system is designed to comply with CJIS v5.1 section 5.4.

Role-based Security

Access to system functionality is restricted based upon security roles. For instance, only a few users need administrative access to the system. This approach reflects the guidelines in CJIS v5.1 section 5.5.2.

Authentication Credentials

HunchLab can delegate credential management to 3rd party directory services such as Active Directory. In such cases, HunchLab assumes that the 3rd party directory service provides a CJIS compliant security model. Additionally, HunchLab can provide a standalone authentication system that complies with both the standard authentication and advanced authentication specifications in CJIS v5.1 sections 5.6.2.1 and 5.6.2.2. Our advanced authentication option provides 2-factor authentication using time-based tokens generated by mobile applications for iOS and Android devices.

Password Management & Login Failures

If operating in stand-alone authentication mode, HunchLab stores user passwords in a salted cryptographic hash format, which increases the computing power necessary to reverse engineer a user's password, even if our database is compromised. Additionally, to prevent external attacks on user credentials, the system keeps track of unsuccessful login attempts and locks the account for progressively longer periods of time. This policy is recommended in CJIS v5.1 section 5.5.3.

Session Lock

When a user logs into HunchLab, a temporary security token is kept within their local browser memory. Upon detecting inactivity for 30 minutes, HunchLab locally encrypts this token using a simple passcode that the user specifies and masks the screen contents. This approach is similar to the way a screen saver masks the screen of a desktop while allowing a user to rapidly access the system again as specified in CJIS v5.1 section 5.5.5.

Data Protection

The HunchLab service is hosted within Amazon Web Services (AWS) data centers. These data centers implement state-of-the-art security practices that protect the physical access to data within HunchLab as recommended in CJIS v5.1 section 5.9. Additionally, AWS continuously monitors their infrastructure against denial of service attacks and penetration vulnerabilities.

Within the HunchLab architecture, Azavea has utilized several security features of the AWS platform to harden the system. For instance, all inbound traffic to HunchLab is encrypted via SSL and terminates at a set of load balancers. These load balancers only allow secure HTTPS traffic and proxy all traffic to the application. Each component of the application is isolated from all others with only the minimum required network traffic for each server instance granted. This security is enforced as inbound and outbound firewall rules on each server, as well as redundantly at the network level. All data in transit within the application is en-

crypted. These design approaches seek to conform to CJIS v5.1 section 5.10.

Personnel

Upon request, Azavea will cooperate with the screening of Azavea personnel with access to the HunchLab system in line with CJIS v5.1 section 5.12.

CHAPTER 5

About Azavea

We believe in the power of technology to improve communities. To reach this goal, we apply geospatial technology to promote civic and social impact in elections, ecosystems, history, public safety, climate change, and more.

Azavea is an award-winning geospatial software design and development company based in Philadelphia. The firm was organized in 2000 to create technologically advanced solutions for web and mobile geospatial data visualization and analysis. Azavea is a certified B Corporation, a for-profit corporation with a social mission. Our mission is to apply geospatial data and software to create more sustainable, vital and livable communities while advancing the state-of-the-art through research. Azavea provides a range of services that include:

- Web and mobile software development
- User interface and experience design
- Mapping and spatial analysis
- High performance computing
- Spatial data mining and modeling
- Research and development

The firm has designed and implemented geographic data applications for a variety of domains including: economic development, elections, urban forestry, crime analysis, humanities and land conservation.

Technology and Partners

Azavea's developers work with a broad range of tools and have particularly strong backgrounds with the .Net, Java, Python, Django and Scala frameworks.

Azavea is an Esri Business Partner and has several years of experience with development and deployment on the ArcGIS platform with dozens of applications implemented on ArcGIS Server and ArcGIS.com. Azavea was named ESRI Business Partner-of-the-Year or Foundation Partner-of-the-Year in 2006, 2007 and 2010. In addition, Azavea is a Microsoft business partner with substantial experience developing the .Net Framework, SQL Server and Windows Server platforms.

Azavea has also partnered with The Omega Group to integrate the predictive missions capability of HunchLab 2.0 in the CrimeView suite of software.

In addition to commercial toolkits, Azavea staff is experienced creating web software solutions that use online API's such as GoogleMaps, Bing Maps, ArcGIS Online and OpenStreetMap. The firm works with a range of open source tools that accelerate and lower the cost of our software development work. In particular, Azavea has a great deal of experience with creating solutions that bring together the strengths of both commercial and open source toolkits to create high quality and visually attractive applications. The firm not only has experience with open source solutions, but also contributes to them, including significant contributions to OpenLayers and PostGIS. In addition, two of Azavea's software solutions, DistrictBuilder and OpenTreeMap, are open source and we release many other software libraries under an open source license.

User Experience Design

Azavea takes great pride in the development of user interfaces that are simple, easy-to-use and are crafted for the specific purpose at hand. Our talented developers and designers work with each client to develop applications that aren't simply functional, they are simple and beautiful.

Commitment to Community

Azavea is committed to working on projects with a strong social value component in order to promote the emergence of more dynamic, vibrant, and sustainable communities. Each of Azavea's projects, products and pro bono engagements showcases this commitment. Azavea works with a range of open source tools that accelerate and lower the cost of our software development work. In particular, Azavea has a great deal of experience with creating solutions that bring together the strengths of both commercial and open source toolkits to create high quality and visually attractive applications. The firm contributes to several open source projects, including the OpenLayers, PostGIS, FastDAO, DistrictBuilder and SourceMap.

Azavea R&D

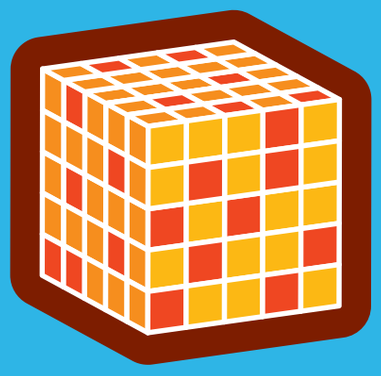
Azavea has an active research and development program through which the firm invests substantial resources toward the development of new solutions and techniques. Each employee is encouraged to develop a personal research project that will both engage the employee and extend the capabilities of the organization. Current research projects include: crime risk forecasting solutions; smart phone applications; cloud computing; creating tools for assessing walkability; and development of an historic geocoder that can support mapping of historic addresses. While not all of these research projects results in measurable commercial success, they are an important part of a culture at Azavea that encourages and takes pride in innovative applications of geospatial technology.

Certified

B

Corporation™

bcorporation.net



HunchLab

The Future in Sight

www.hunchlab.com
info@azavea.com
215.925.2600