Supplementary Materials

S1. *Additional Method Details*

Table S1. **Legend of mobility trace feature names to abbreviations.**

Abbreviation	**Name**
VAR	Location variance
AVG SPD	Average moving speed (km/h)
ENT	Entropy
NORM ENT	Normalized entropy
HOME	Time spent at home
TRANS TIME	Transition time
TOT DIST	Total distance travelled
ROUT IND	Routine index
INDGR	Indegree
OUTDGR	Outdegree
UNIQUEC	Number of unique locations visted
UNITC	Unique cluster type
OUTDOORS_REC	‘Outdoors & Recreation’ cluster
PROFESS_OTH	‘Professional & Other Places’ cluster
SHOP	‘Shop & Service’ cluster
FOOD	‘Food’ cluster
TRANSPORT	‘Travel & Transport’ cluster
RESIDENCE	‘Residence’ cluster
UNIVERSITY	‘College & University’ cluster
ARTS_ENTERT	‘Arts & Entertainment’ cluster
NIGHTLIFE	‘Nightlife Spot’ cluster

S1.1. *Hybrid sample algorithm*

Algorithm S1 Algorithm for generating hybrid dataset

```
**Input**: Number of hybrid samples with a *case* status (n_cases), Number of hybrid samples with a *control* status (n_controls), target number of case and control samples ($n$), Genotype data and labels ($\boldsymbol{X}^g, \boldsymbol{y}^g$), mobility trace data and labels ($\boldsymbol{X}^m, \boldsymbol{y}^m$), Comorbidity $C$, Mobility trace relative risk $M$, Genotype relative risk $G$
**Output**: hybrid_data ($\hat{\boldsymbol{X}}, \hat{\boldsymbol{y}}$)
while n_cases or n_controls $\leq n$ do
    $\hat{\boldsymbol{x}}_i^g \sim s(\mathcal{U}(0,1))$                    ▷ Select case or control genotype at random
    $\hat{\boldsymbol{x}}_i^m \leftarrow$ simulate_mt_sample* according to $C$
    $\mathbb{1}(\hat{\boldsymbol{x}}_i) \leftarrow 1$ if $\boldsymbol{x}_i^g$ is a case 0 otherwise
    $a^g \sim \mathrm{Bern}(G/(G+1))$
    $a^m \sim \mathrm{Bern}(M/(M+1))$
    hybrid_sample $\leftarrow$ concatenate($\hat{\boldsymbol{x}}_i^g, \hat{\boldsymbol{x}}_i^m$)
    $\hat{y} \leftarrow \mathbb{1}(\hat{\boldsymbol{x}}_i^g) \cdot a^g \vee \mathbb{1}(\hat{\boldsymbol{x}}_i^m) \cdot a^m$
    hyrbid_data.push(hybrid_sample, $\hat{y}$)
end while
```

Algorithm S2 simulate_mt_sample*

```
**Input**: $\boldsymbol{X}^m$
**Output**: synth_sample
$N^m \leftarrow$ number of mobility trace samples
$L^m \leftarrow$ number of features
for $j$ in $L^m$ do
    feature_vect $\leftarrow \boldsymbol{v}^m_{N^m,j}$
    $f(\cdot) \leftarrow$ Inverted ECDF of feature_vect
    $a \sim \mathcal{U}(0,1))$
    synth_sample.push($f(a)$)
end for
```

S2. *Additional Results*

Table S2. **PR AUC scores across models and feature sets.** Average PR AUC and standard errors across 100 datasets with comorbidity $C = 0.8$ and relative risks $(G,M) = (\infty,\infty)$. Bold values indicate models with significantly better PR AUC than other methods for a feature set (Welch's two sample t-test; all p-values $\leq 2.39 \times 10^{-8}$).

		Data set	
Model	**Merged**	**Mobility Trace**	**Genotype**
Logistic Regression (LOGIT)	0.67 ±0.03291	0.64 ±0.03039	0.60 ±0.03009
Linear SVC (SVC)	0.67 ±0.03287	0.64 ±0.03040	0.60 ±0.03007
K-Nearest Neighbors (KNN)	0.68 ±0.03333	0.71 ±0.03398	0.62 ±0.03119
Decision Tree (DT)	0.75 ±0.03602	0.76 ±0.03588	0.61 ±0.03066
Random Forest (RF)	0.91 ±0.04320	0.91 ±0.04290	**0.65 ±0.03242**
AdaBoost (ADA)	0.91 ±0.04366	0.91 ±0.04329	0.61 ±0.03244
Gradient Boosting Classifier (GBC)	**0.93 ±0.04462**	**0.93 ±0.04413**	0.64 ±0.03205

Table S3. **Correlations between SHAP and feature values.** Pearson's correlation for SHAP and feature values for random forest and SVC models. *p-value$\leq 2.2 \times 10^{-16}$

Model	Random Forest	Support Vector Classifier
VAR	0.5*	-0.88*
AVG SPD	-0.17*	0.86*
ENT	0.45*	-0.79*
NORM ENT	0.17*	-0.61*
HOME	-0.16*	0.59*
TRANS TIME	-0.50*	0.80*
TOT DIST	0.54*	0.62*
ROUT IND	-0.15*	-0.14*
INDGR	0.29*	-0.75*
OUTDGR	0.24*	0.59*
UNIQUEC	0.05*	0.54*
UNITC	0.02*	-0.09*
OUTDOORS_REC	-0.30*	0.81*
PROFESS_OTH	-0.74*	0.97*
SHOP	0.39*	-0.92*
FOOD	0.23*	-0.60*
TRANSPORT	0.15*	-0.76*
RESIDENCE	0.09*	-0.82*
UNIVERSITY	0.25*	-0.37*
ARTS_ENTERT	0.29*	-0.86*
NIGHTLIFE	0.05*	-0.70*
RS12130499_A	0.60*	-0.88*
RS12712037_A	-0.51*	0.89*
RS1491583_G	-0.68*	0.93*
RS2443067_G	-0.61*	0.91*
RS1322444_A	-0.75*	0.97*
RS3798683_A	-0.71*	0.95*
RS17770427_A	-0.72*	0.94*
RS10504659_A	-0.73*	0.94*
RS16939567_C	-0.69*	0.94*
RS1554347_G	-0.70*	0.94*

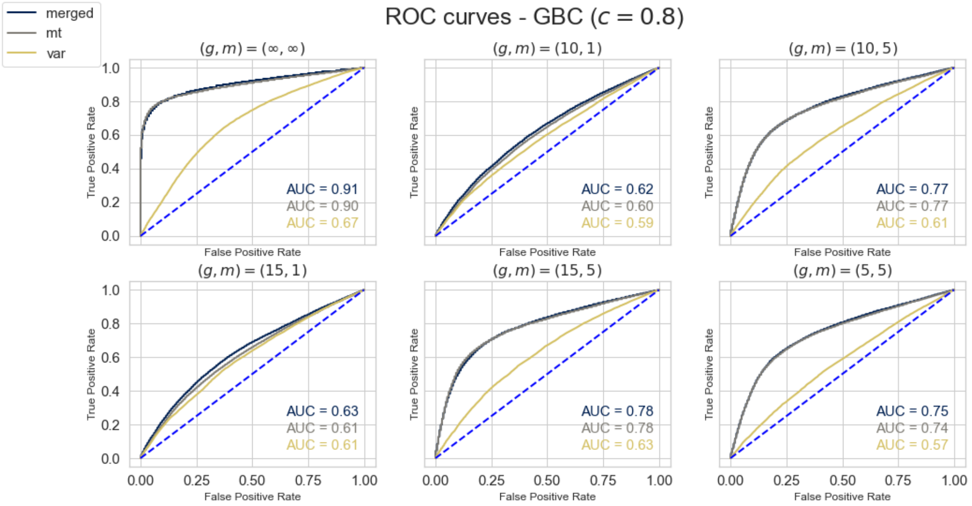

Fig. S1. **ROC curves across relative risk configurations.** GBC models were trained on 100 randomly sampled datasets across relative risks $(G,M) \in \{(\infty,\infty),(10,1),(10,5),(15,1),(15,5),(5,5)\}$.

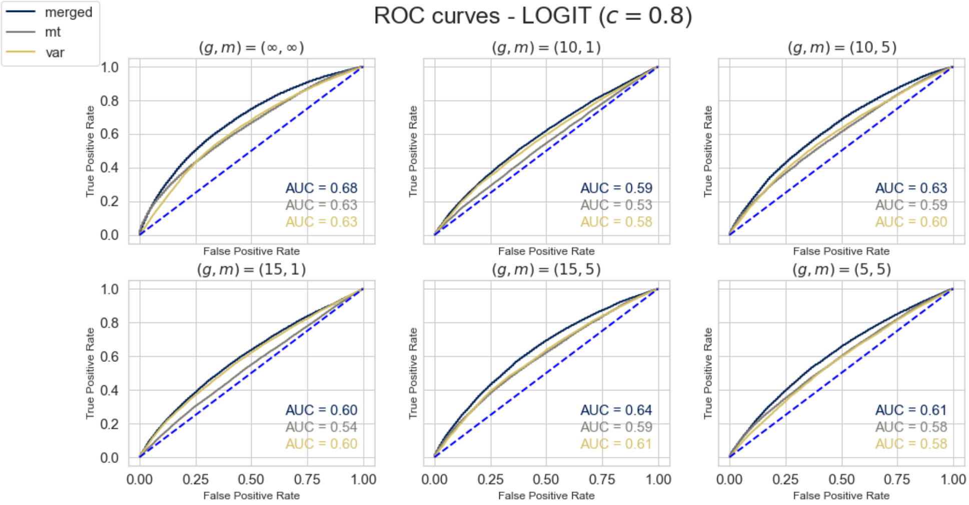

Fig. S2. **ROC curves across relative risk configurations.** Logistic regression models were trained on 100 randomly sampled datasets across relative risks $(G,M) \in \{(\infty,\infty),(10,1),(10,5),(15,1),(15,5),(5,5)\}$.

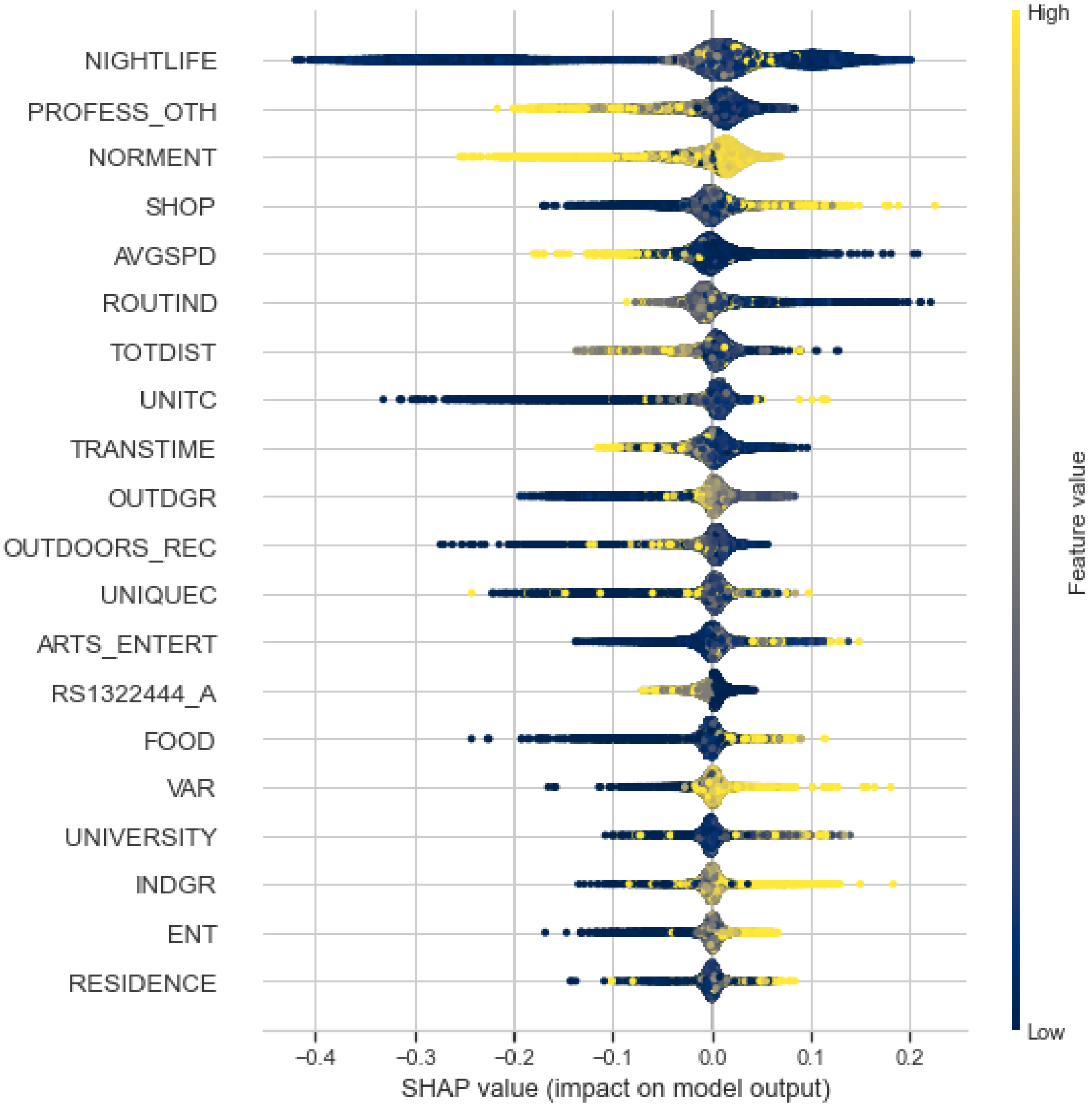

Fig. S3. **SHAP value (x axis) and feature value (color) are shown across the 20 features with the largest mean difference between expected and predicted values for RF. SHAP values are computed from 100 datasets with comorbidity $C = 0.8$ and relative risks $(G,M) = (10,5)$.**

S3. *Data Processing*

Genetic Data Processing. We processed the genetic data using PLINK[31] by removing duplicate features and variants that (a) had minor allele frequency less than 0.01, (b) violated Hardy-Weinberg Equilibrium (χ^2 test; p-value less than 10^{-7}), or (c) were missing in more than 1% of the samples. Samples were then removed if they had more than 1% missing variants, reported sex did not match inferred sex, or identified as having a relative in the data. Missing alleles were then imputed by their mode. Because most of the reference variants could not be extracted from the heroin dependence genotype samples, linkage disequilibrium was leveraged to use variants in the same region as the reference variants were used as proxies. Using the Ensembl REST API, proxy variants were retrieved from all African and European subpopulations in the 1000 GENOMES project, phase 3 (D_prime=1.0, window size=300kb, and $R^2 = 0.8$). Fetched variants were then used in PLINK to extract the genetic data from the opioid genotype dataset.

Mobility Trace Data Processing. For mobility trace data, we identified discrete locations an individual has visited by running the density-based clustering algorithm DBSCAN with hyperparameters selected via grid search (Supplemental Results). We removed points where individuals are identified as travelling faster than 10 kilometers per hour, since this likely occurs in automobile transit and can create spurious clusters. Distinct clusters were labelled using the Google Places API, which includes bounding boxes for discrete places and 9 categories: Outdoors and Recreation, Professional & Other Places, Shop & Service, Food, Travel & Transport, Residence, College and University, Arts & Entertainment, and Nightlife Spots. If a cluster centroid is not within the bounding box of any known place, we match it with the bounding box closest to the cluster centroid. We then generated the mobility feature matrix $\boldsymbol{X}^g$ by computing the aforementioned mobility features ($L^m = 21$).

S4. *Model selection*

Since we observed collinearity in the genetic data that precluded fitting linear models, we selected genetic features using backward stepwise regression on a held out dataset for each comorbidity level and RR configuration. We also performed hyperparameter selection for each risk score model and the 3 separate feature sets separately. We used 10-fold CV with an average precision scoring metric and a grid or 60 iteration randomized search depending on the dimension of the hyperparameter search space. We selected DBSCAN hyperparameters ϵ and the minimum number of points to define a cluster using silhouette score. The hyperparameter ϵ was varied from 0.002 to 0.055 in increments of 0.002. The minimum number of cluster points was varied from 2 to 162 in increments of 2.

S5. *Model Interpretation*

We compute model interpretations using the SHapely Additive exPlanations (SHAP), a model agnostic approach based on Shapely values and coalitional game theory for interpreting fitted models and quantifying feature importance.[34] It does so by defining an explanation model g, which is an interpretable approximation of the original prediction model f. The explanation model is defined as $g(z') = \phi_0 + \sum_{j=1}^{M} \phi_j z'_j$ where $z' \in \{0,1\}^M$ selects a subset of features, M is the subset size, and $\phi_j \in \mathbb{R}$ is the feature effect attribution for a feature j.[34] As a result, this additive feature attribution method produces *SHAP values*, a *post hoc* measure of feature importance which are approximated by various methods. SHAP values for a data instance x attribute to each feature the change in the expected model prediction when conditioning on that feature. This change is the difference between the expected value $\mathbb{E}[f(z)]$ that would be predicted if all features to the current output $f(x)$ were unknown.

We used TreeSHAP to explain our tree-based models, and chose the `auto` algorithm parameter for the single model explainers to optimize training time. Because AdaBoost and K-NN were not natively supported by `shap v.0.39`, we have not yet generated explanation results for those methods.

SHAP summary plots combine feature importance with feature effects. Each point in the plot represents a Shapely value for a feature and an instance. The y-axis presents the features ordered according to their average contribution, and the position on the x-axis is determined by the Shapely (SHAP) value. The color of each point indicates the actual feature value. SHAP dependence plots are scatter plots containing the following points $\{(x_j^{(i)}, \phi_j^{(i)})\}_{i=1}^{n}$, with $x_j^{(i)}$ on the x-axis and $\phi_j^{(i)}$ on the y-axis. As opposed to accumulated local effects and partial dependence plots, SHAP dependence plots also account for the interaction effects present in the features. An interaction effect is the additional combined feature effect after accounting for the individual feature effects.[38] The vertical dispersion of SHAP values at a single feature value is driven by interaction effects, and another feature is chosen for coloring to highlight possible interactions.