Estimating Geographic Origin of Human Populations using Genomic Data

Hosein Rajabi
Department of Computer Science
Western University
hrajabi@uwo.ca

***Abstract*—Local ancestry inference (LAI), also referred to as ancestry deconvolution provides the comprehensive analysis of biological adaptation and phenotypic variations at different regions along the human genome. LAI is an essential part of the current medical advances in DNA sequencing with applications extending from identifying genetic susceptibility loci under natural selection to polygenic risk scores (used to predict traits in embryos and disease risk in adults), inference of human population history of origin and solving legal cases relevant to genetic associations.**

The current industrial DNA tests e.g. 23andMe mostly use random forest (RF) and support vector machines (SVM) techniques to provide only a continent-level inference of ancestry estimation, even when using millions of features along a DNA. The focus of this study is to develop machine learning techniques for ancestry deconvolution, with accuracy comparable to the current models as well as robustness to missing data. We are particularly interested in going further than continent-level classification of global ancestry deconvolution by estimating distinct subgroups within each continent.

This project uses a dataset with a total of 1986 chromosomes taken from single-ancestry populations referring to two main clusters; South Asia and East Asia, where each of them has 1812841 features specifying distinct biological traits which makes an individual unique. After doing prepossessing and feature engineering on the dataset, we use several supervised learning algorithms to come up with the exact geographic location of the genomic origin as a (longitude, latitude) pair. A test set accuracy of 92.33% is achieved using several methods including Gaussian process regression.

I. Introduction

The human genome is composed of chromosomes. Each chromosome contains the genetic information that encodes the instructions for the development and function of an organism. This information is stored in the form of deoxyribonucleic acid, or DNA, which is a long, double-stranded molecule made up of nucleotides. There are four types of nucleotides: adenine (A), thymine (T), guanine (G), and cytosine (C). The sequence of these nucleotides along the DNA molecule determines the genetic code that specifies the traits and characteristics of an organism.

Full-genome sequencing takes the whole variation along the DNA molecule, and reports all of the DNA information, 3 billion base pairs in the case of humans, which are about 99.5% identical from person to person. That is, much of the human genome does not vary between individuals, so they are redundant data. However, there are certain positions along the human genome (about 0.5% percent of the total positions), called single nucleotide polymorphisms (SNPs), that are known to vary within the population (Fig. 1) and are linked to health conditions, traits and ancestry groups that make a unique individual.

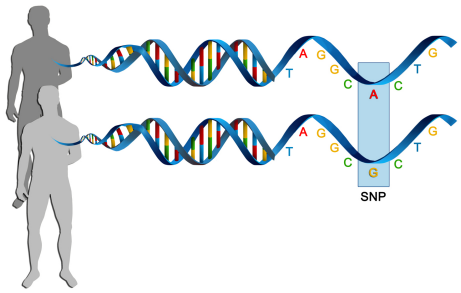

Fig. 1: Discernible SNP data in different individuals

Because DNA is inherited as an intact sequence with only rare, random swaps between the two parental DNA sequences at each generation, ancestral SNPs allow for powerful ancestry inference in the admixed individuals as well.

In humans, each cell normally contains 23 pairs of chromosomes, for a total of 46. Twenty-two of these pairs, called autosomes, look the same in both males and females. The 23rd pair, the sex chromosomes, differs between males and females. In this study, we are following the available dataset and not considering the sex chromosomes, as its inclusion would add significantly more complexity to the model.

The input to our algorithm is the one-hot encoded version of "ACGT" letters along the one-dimensional DNA sequence. The input after doing data preprocessing and dimension reduction is a matrix of size 1786×1630 for the training set. The labels are (latitude, longitude) pairs obtained from Google Maps, at the center of 10 different cities in East and South Asia. We then use several supervised learning methods including Gaussian process, Support vector regression, Linear regression, Multi-layer perceptron, Random forest, Decision tree, and so on to train the model and then test it on a test set of size 200×1630 (the feature size after preprocessing and feature

reduction is 1630 in both train and test sets). Consequently, we come up with the (latitude, longitude) pair representing the biogeography of worldwide individuals down to their country of origin.

II. Existing Methods

Recent advances in machine learning techniques and the growth in number of available genomics datasets from DNA tests, are facilitating the identification and classification of ancestry-based patterns with an increasing accuracy. The ability to accurately infer the ancestry along the genome in high-resolution is important to identify the role of genetics and environment for complex traits including disease-susceptible loci. So far, different LAI methods have been developed.

Hidden Markov Models (HMMs) have been employed to model local-ancestry correlations in SABER [1], HAPAA [2] and HAPMIX [3]. The LAMP algorithm used in [4] utilizes probability maximization within a sliding window providing better and faster performance than several HMMs-based models, even in admixed individuals. RFMix [5] is a discriminative model that uses conditional random fields (CRF) based on random forests within windowed sections of the genome. This method has shown to be both faster and more accurate (95.2% accuracy) than LAMP and previous HMMs-based methods.

While these models have proved to work well with the continent-level classification of different ethnic groups, they don't address the sub-continent level estimations with a reasonable accuracy and speed. This has motivated another category of studies focused on sub-continent populations.

Elhaik et al. [6] have used the Geographic Population Structure (GPS) algorithm and demonstrate its accuracy with three data sets using $40,000 - 130,000$ SNPs with an accuracy of 700km in Europe. However, their models were inaccurate elsewhere.

Novembre and Peter [7] have studied fine-scale population structure in humans. They improved statistics and models for better capturing differentiation, admixed descendants, and the spatial distribution of variation; computational speed-ups that allow methods to scale to modern data. However, the limitations of discrete population models, uncertainty in individual origins, the incorporation of both fine-scale structure and ancient DNA in parametric models, and the development of efficient computational tools are the limitations in their study.

Consortium et al. [8] have used SNP data in Europe and have mapped the genetic variations to two main dimensions using PCA. They have characterized genetic variation in a sample of 3,000 European individuals and have found a close correspondence between genetic and geographic distances. Their model performs separate predictions on longitude and latitude. However, it doesn't provide a prediction for the other continents.

We believe this former study is the most relevant and most rigorous one in terms of providing a prediction for the sub-populations within a continent. That being said, our goal is trying to come up with such an estimation which is fast and accurate, and is capable of predicting different populations withing each continent along the world. In this work we have focused on Asia continent due to the limited dataset made available to us. A potential future study to improve this research would be adding datapoints from Europe and Africa as well.

III. Dataset and Features

A. Dataset

In this work, we used SNP dataset from Department of Biomedical Data Sciences at Stanford Medicine, where their dataset is originally obtained from human research participants through the 1000 genomes project [9]. The dataset comprises of single-ancestry individuals (who have DNA strands all from the same region) used for both training and test purposes. For the future validation purposes, we will use admixed descendants (who have DNAs from a distantly-related population or species, as a result of interbreeding between populations or species who have been reproductively isolated and genetically differentiated) for testing purposes to make our model more rigorous.

The SNP dataset of the training set had a total of 1786 single-population individuals from 10 different cities in two main clusters: South Asia (purple dots in Fig.2) and East Asia (green dots in Fig.2). The dataset is balanced as the 1786 individual data comprises of roughly 180 individuals from each of the 10 cities. Their corresponding populations are listed in the table below. The (latitude, longitude) pairs obtained from Google Maps for these 10 cities serve as our labels. However, in order to geographically discretize our dataset, we considered non-overlapping squares centered at each of these dots in such a way that it approximately covers the whole area of that particular city. Subsequently, we assign all the input DNA sequences that fall inside one square to the city represented by the center of that square.

Furthermore, each set of chromosome in our dataset has 1812841 features. These features define the unique traits/ health conditions between humans, e.g. eye/hair color, round/oval shaped face, diabetes issues inherited through genetics and so on. Needless to say, it' is very undesirable and computationally exhausting to build models for such a high-dimensional feature space. Consequently, data preprocessing and feature selection using dimension reduction techniques is essential.

B. Feature Selection

The main challenge in analyzing human's genome arises from the dataset being very high-dimensional. For this reason, we did data normalization and then applied Principal Component Analysis as well as Variance Threshold methods to perform

Code	Population	City
CHB	Han in Beijing	Beijing
CHS	Southern Han	Guangzhou
JPT	Japanese	Tokyo
KHV	Vietnamese	Ho Chi Minh City
CDX	Dai	Xishuangbanna
GIH	Gujarati	Gandhinagar
PJL	Punjabi	Lahore
BEB	Bengali	Dhaka
STU	Sri Lankan Tamil	Jaffna
ITU	Telugu	Hyderabad

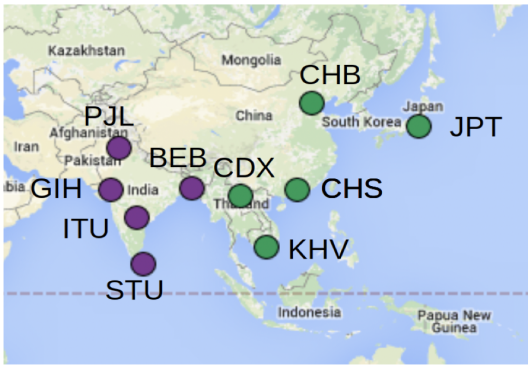

Fig. 2: Top: The ten different cities of our study and their corresponding populations. Bottom : centers of the datapoints, the (latitude,longitude) pairs of which serve as our labels

feature reduction. We kept the principal components that would retain 95% of variance of the features. The resulting PCA transformation reduced our feature space to $X_i : (1786 \times 1630)$. As we can see in the scree plot in Fig.3, we can't reduce the features any further as all the 1630 features seem to be important. When applying PCA, it is crucial to train the PCA model on the training dataset only and then apply the same transformation to both the training and test datasets. This ensures that the test data is transformed in the same way as the training data and that the model is not overfitting to the test data.

In Fig.3, the first two principal components plotted for all data points on the top plot represents that East/south Asia are well separated. The third plot is schematizing PJL and CHS, which are again well separated. However, when we plot them for CHS and CHB (the fourth plot), they seem to be overlapping.

IV. Methods

We have tried several supervised learning regression methods using Sklearn libraries. Fig.4 in the Results section summarizes the performance of different methods on our dataset. As working with almost 2 million features was impractical, we reduced the number of features into 1630 using PCA dimension reduction. Variance threshold feature selection was applied for this purpose as well, however, its performance was not better than PCA, therefore we only used

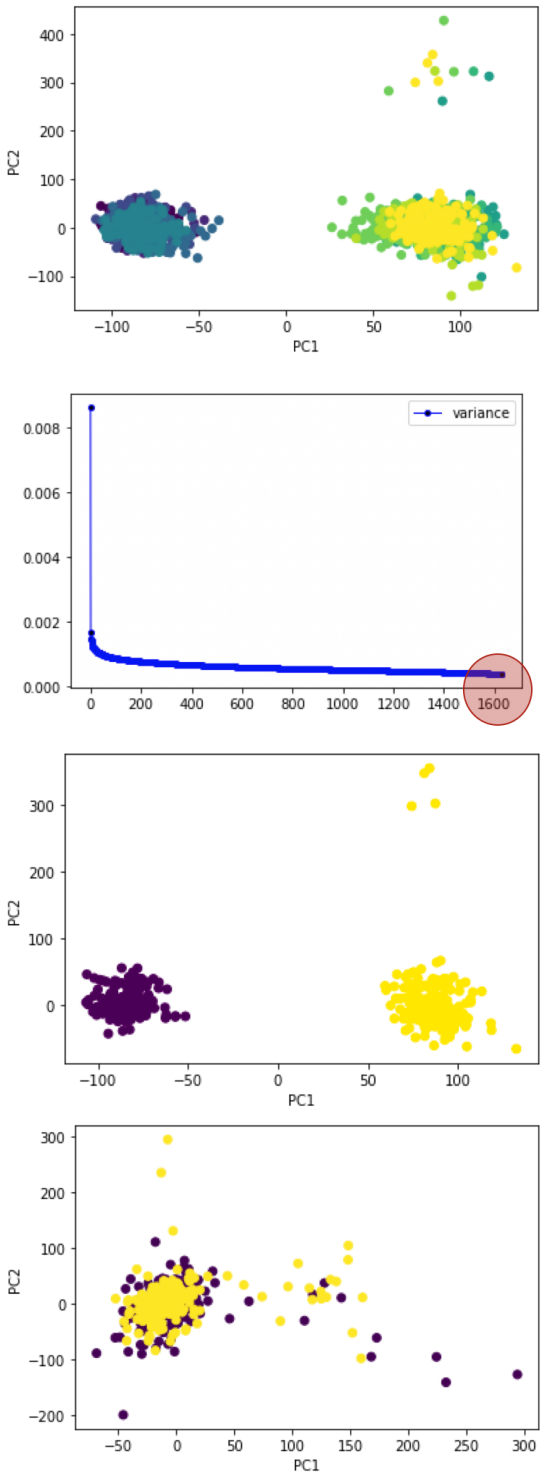

Fig. 3: Principal Component Analysis on the sub-populations; first: all ethnic groups, second: scree plot of principal components, third : PJL and CHS, fourth: CHB and CHS

PCA for dimension reduction before feeding the data into models.

Linear Regression fits a model by using ordinary least

squares and minimizes the residual sum of squares between the ground-truth targets and the predicted targets, i.e. $\min_w ||Xw - y||_2^2$. In order to reduce model complexity and prevent over-fitting of linear regression, we implemented the regularized versions of it through Ridge Regression $\min_w ||Xw-y||_2^2+\alpha||w||_2^2$ with different complexity parameters $\alpha = 0.01, 100$, as well as Lasso regression with the objective function $\min_w ||Xw - y||_2^2 + \alpha||w||_1$ with $\alpha = 1, 0.01, 0.0001$. However, these regularized alternatives didn't seem to shrink the coefficients much. Therefore, we are just reporting the unregularized version on the table.

Gaussian Process Regression is based on Bayesian approach (posterior $= \frac{\text{likelihood} \times \text{prior}}{\text{marginal likelihood}}$). It calculates the probability distribution over all the functions that can fit the data. We need to specify a prior (on the function space), calculate the posterior using the training data, and compute the predictive posterior distribution on our points of interest.We used the RBF kernel $K(x, z) = exp(-\gamma||x - z||^2)$ and random state=138.

Automatic Relevance Determination fits regression model with Bayesian Ridge Regression where the coefficient weights are shifted to zero to stabilize the model. We computed the objective function at each step. The objective function used in ARD is a combination of the likelihood function and a prior distribution on the coefficients. At each step of the algorithm, the objective function is computed to evaluate the relevance of each feature. Features with coefficients that are less relevant to the task at hand will be assigned a small weight and will be removed from the model. The shape parameter for the Gamma distribution (α_1) and the inverse scale parameter (α_2) are hyperparameters in the ARD model. These parameters control the degree of regularization and the sparsity of the resulting model. The shape parameter for the Gamma distribution $\alpha_1 = 1e - 6$ and inverse scale parameter $\alpha_2 = 1e - 6$.

Bayesian Ridge provides predictions by iteratively maximizing the marginal log-likelihood of the observations. We took regularization parameters for the coefficients as the previous case.

Huber Regression applies a linear loss to samples that are classified as outliers. It does not ignore the effect of the outliers but gives a lesser weight to them. Its loss function is $\min_{w,\sigma} \sum_{i=1}^{n} \left(\sigma + H_\epsilon\left(\frac{X_i w - y_i}{\sigma}\right)\sigma\right) + \alpha||w||_2{}^2$ where $H_\epsilon(z) = \begin{cases} z^2, & \text{if } |z| < \epsilon, \\ 2\epsilon|z| - \epsilon^2, & \text{otherwise} \end{cases}$. The hyperparameters are $\alpha = (1e - 2, 1e - 4, 1e - 6)$, $\varepsilon = (1.1, 5, 10, 20)$, Max iter = (50,100), where the last number in each set gave the best results.

Bagging Regression works based on introducing randomization into the prediction procedure. It is an ensemble meta-estimator that fits models on the random subsets of the base model and then averages out the results to give a final estimation. The random subset selection helps in reducing the variance, e.g. in tree-based schemes. We used this model with a linear regression as the base estimator and random state=138.

Multi-layer Perceptron can learn a non-linear function for the regression task by incorporating several non-linear hidden layers between input and output layers. It traines through back-propagation and optimizes the squared loss function using LBFGS or stochastic gradient descent. It's sensitive to feature scaling and random weight initializations can lead to different validation accuracy. We trained with hyperparameters Adam and lbfgs optimizers, $\alpha = 1e - 5$, Hidden layer size: (3,5,3) and random state=1.

Random Forest Regression is another meta-estimator that works by fitting several decision-trees on random subsets of the dataset and taking the average to improve the accuracy. This way it can control overfitting as well. Our samples are drawn with replacement, the number of estimators=500, Max depth=5, min sample splits=2, random state=138.

Support Vector Regression with stochastic gradient descent to minimize hinge loss was used. Polynomial kernel of 2nd degree gave better results than other kernels. Best regularization parameters were (c=10), $\gamma = 10$, $\varepsilon = 0.1$.

Decision Tree Regression predicts the value of a target variable by learning simple decision rules inferred from the data features. We tried Max depth=(4,5,7) where 5 gave the best results.

Gradient Boosting Regression with number of estimators = 100, Max depth = 5, learning rate= 0.01, min sample splits = 2, and least square loss was tried, but didn't provide a good approximation. Kernel Ridge with linear kernel and $\gamma = 0.8$ was tried as well. Adaboost Regression with number of estimators = -100 and random state=138 was tried as well. However, they were not among our good models, so for the sake of brevity, we don't go into details about them here.

V. Results and Discussion

The feature/example ratio is high in this project, therefore cross validation (k = 5) is adopted to find the best hyperparameters. For tuning the hyperparameters of our model, we did a random grid search on each model and chose the best parameter pair that gives the lowest CV error, this has been achieved by means of the RandomizedSearchCV module of scikit-learn and resulted in the hyperparameters listed in the methods section for each model. This helps to avoid overfitting and ensures that the model generalizes well to new, unseen data. The process of cross validation involves splitting the data into k subsets, training the model on k-1 subsets and evaluating it on the held-out subset, repeating this process k times and

Models	R^2	RMSE	MAE	R^2(train set)	conf. intervals
Linear Regression	(0.9233, 0.5124)	(5.70 %,46.31 %)	(4.66 %,30.65 %)	(0.973, 0.5824)	(0.0276, 0.0325)
Gaussian Process Reg	(0.9233, 0.5124)	(5.70 %,46.31 %)	(4.66 %,30.65 %)	(0.973,0.5824)	(0.92883,1.3025)
Automatic Relevance Det	(0.9096, 0.4774)	(6.11 %,49.10 %)	(4.99 %,31.96 %)	(0.969,0.5874)	(0.9127,0.9935)
Bayesian Ridge	(0.9182, 0.4776)	(5.83 %,48.25 %)	(4.78 %,31.83 %)	(0.968,0.5376)	(0.2838,0.4231)
Huber Regression	(0.9233, 0.5124)	(5.70 %,46.31 %)	(4.66 %,30.65 %)	(0.973,0.5824)	(0.24656,0.29022)
Bagging Regression	(0.9106, 0.4436)	(6.04 %,49.67 %)	(4.92 %,32.90 %)	(0.9493,0.5436)	(0.7556, 0.8861)
Multi-layer Perceptron	(0.9233, 0.5124)	(5.70 %,46.31 %)	(4.66 %,30.65 %)	(0.9733,0.5724)	(0.9187, 1.3124)
Random Forest Regression	(0.9114, 0.4059)	(6.04 %,48.87 %)	(4.63 %,33.12 %)	(0.931,0.4659)	(0.9288,1.3025)
Support Vector Regression	(0.5107, 0.2407)	(16.96 %,58.15 %)	(11.83 %,38.21 %)	(0.53,0.29)	(0.02341,0.0812)
Decision Tree Regression	(0.8895, 0.3951)	(6.74 %,52.03 %)	(4.72 %,33.85 %)	(0.898,0.415)	(0.7706,1.513)
Gradient Boosting Reg	(0.7879, 0.3489)	(9.85 %,53.24 %)	(8.41 %,35.24 %)	(0.7979,3689)	(0.7706,1.51295)

Fig. 4: Results from different algorithms

averaging the performance metrics across all k runs. By doing so, we are able to estimate the model's performance on new unseen data and select the hyperparameters that produce the best overall performance.

The results are reported in Fig. 4 for (longitude, latitude) pairs. All the columns except the last one show the performance on the test set.

As depicted in fig. 5, these errors suggest that regression models are successfully learning to identify longitudes. The models predict longitude with much better accuracy. The predictions for latitudes have lower accuracy, because different ethnic groups change significantly along the longitude, while small changes happen along the latitude. For example for CHB and CHS, the genetics information is very similar, but there is a significant difference between their latitude.

Multi-layer Perceptron (MLP), Hubber Regression (HR), Logistic Regression (LR), and Gaussian Processes (GP) did almost the same on both latitude, longitude predictions. Support Vector Regression (SVR) had the worst prediction. Among the best models, GP and MLP give the best prediction, since they have the highest confidence intervals and LR has the lowest.

The confidence intervals reported for the training data in the last column show the variance over the reported longitude/latitude (e.g. longitude = predicted longitude$\pm\sigma$). Based on what we observe, our models do not give a high-confidence interval for the predicted outputs. The units of latitude, longitude are in degree.

It should be noted that we also tried multi-output models instead of separate latitude/longitude training and the reported error is the average of what we get from separate training on each of them. However, we chose the current approach as it gives us a better insight about the source of inaccuracy. One other approach we will try in the future is doing multi-task learning for the latitude prediction in conjunction with the longitude prediction tasks using neural networks. Multi-task learning works based on parameter sharing (parameters of neural net being shared between different output tasks). Parameter sharing is advantageous as the tasks are learnt simultaneously and the gradient updates from the two loss functions inform both outputs, resulting in a more generalizable model. This could be done using the hard parameter sharing (when the hidden layers are shared between all the output tasks) or the soft parameter sharing (each label has its own neural net with its own parameters, and the parameters of different models are encouraged to stay similar using a regularization parameter).

VI. Conclusion and Future work

Even though we get good accuracy for the longitude predictions, the latitude accuracy and confidence intervals can be improved by using a larger and more diverse dataset. For example, we believe that adding datapoints from Europe and Africa regions would most probably improve the predictions, this is because unlike East Asia where the country is very large and our data points are very distant while genetic variations were not considerable, in Europe populations with more genetic distinctions are well distributed in closer cities/villages. However, this needs further investigation on the new dataset.

In order to increase size of our dataset and its genetic complexity, we could use the Wright-Fisher forward simulation to generate more data for the admixed descendants by leveraging the available full genome single-ancestry data. Training a Convolutional Neural Network or Recurrent Neural Network by incorporating more diverse data points of both single-population and admixed descendants from well-separated regions in the train set would be done as well. Feeding the results of the trained CNN/RNN into Random Forest would be tried as an accuracy booster.

We can also model this problem using Graph Neural Networks by encoding the features into the nodes of the graph and then training the model by encouraging the similar ethnic groups to stay closer to each other in the embedding space

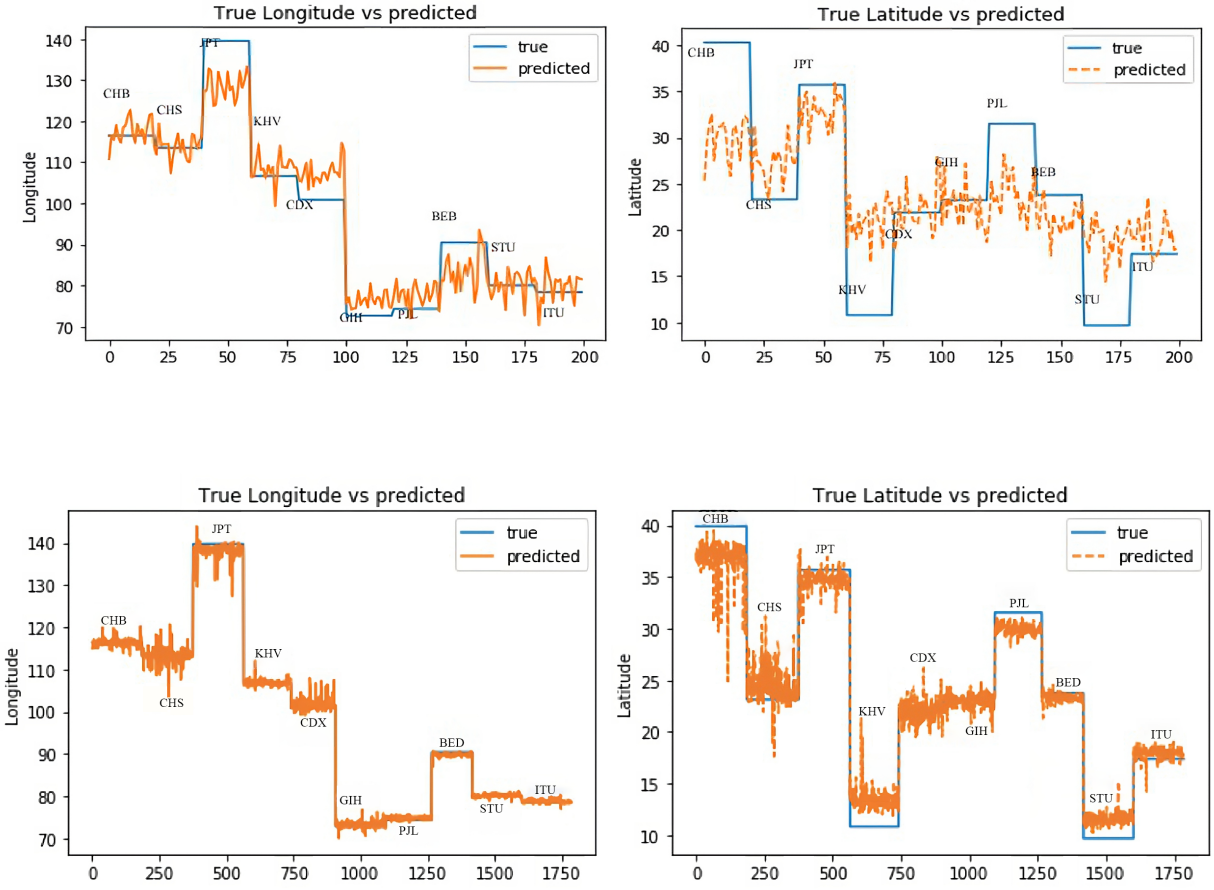

Fig. 5: Top: The true values vs. predicted results from our best model (GP) for latitude and longitude represent a reasonable prediction. Bottom : Sanity check for the training set labels vs predicted

through the edges of the graph. However, this would be more computationally demanding than the above-mentioned models.

In terms of practical applications, improving the accuracy of these models could have significant implications for personalized medicine, such as predicting the risk of certain diseases based on genetic ancestry, or optimizing drug treatments based on individual genetic makeup. Additionally, this work could contribute to a better understanding of human migration patterns and historical relationships between different populations.

References

[1] H. Tang, M. Coram, P. Wang, X. Zhu, and N. Risch, "Reconstructing genetic ancestry blocks in admixed individuals," *The American Journal of Human Genetics*, vol. 79, no. 1, pp. 1–12, 2006.

[2] A. Sundquist, E. Fratkin, C. B. Do, and S. Batzoglou, "Effect of genetic divergence in identifying ancestral origin using hapaa," *Genome research*, vol. 18, no. 4, pp. 676–682, 2008.

[3] A. L. Price, A. Tandon, N. Patterson, K. C. Barnes, N. Rafaels, I. Ruczinski, T. H. Beaty, R. Mathias, D. Reich, and S. Myers, "Sensitive detection of chromosomal segments of distinct ancestry in admixed populations," *PLoS genetics*, vol. 5, no. 6, p. e1000519, 2009.

[4] S. Sankararaman, S. Sridhar, G. Kimmel, and E. Halperin, "Estimating local ancestry in admixed populations," *The American Journal of Human Genetics*, vol. 82, no. 2, pp. 290–303, 2008.

[5] B. K. Maples, S. Gravel, E. E. Kenny, and C. D. Bustamante, "Rfmix: a discriminative modeling approach for rapid and robust local-ancestry inference," *The American Journal of Human Genetics*, vol. 93, no. 2, pp. 278–288, 2013.

[6] E. Elhaik, T. Tatarinova, D. Chebotarev, I. S. Piras, C. M. Calò, A. De Montis, M. Atzori, M. Marini, S. Tofanelli, P. Francalacci, *et al.*, "Geographic population structure analysis of worldwide human populations infers their biogeographical origins," *Nature communications*, vol. 5, p. 3513, 2014.

[7] J. Novembre and B. M. Peter, "Recent advances in the study of fine-scale population structure in humans," *Current opinion in genetics & development*, vol. 41, pp. 98–105, 2016.

[8] J. Novembre, T. Johnson, K. Bryc, Z. Kutalik, A. R. Boyko, A. Auton, A. Indap, K. S. King, S. Bergmann, M. R. Nelson, *et al.*, "Genes mirror geography within europe," *Nature*, vol. 456, no. 7218, p. 98, 2008.

[9] . G. P. Consortium *et al.*, "A global reference for human genetic variation," *Nature*, vol. 526, no. 7571, p. 68, 2015.