PNAS
www.pnas.org

Supplementary Information for

Computational Studies of Anaplastic Lymphoma Kinase Mutations Reveal Common Mechanisms of Oncogenic Activation

Keshav Patil, Earl Joseph Jordan, Jin H. Park, Krishna Suresh, Courtney M. Smith, Abigail A. Lemmon, Yaël P. Mossé, Mark A. Lemmon*, Ravi Radhakrishnan*

Corresponding Author Names. Mark A. Lemmon, Ravi Radhakrishnan
E-mail: mark.lemmon@yale.edu; rradhak@seas.upenn.edu

This PDF file includes:

Supporting Information Text

1. Experimental Methods

A. Plasmid Construction. DNA encoding kinase domain residues 1090-1416 of human ALK was amplified by using primers that included an N-terminal hexahistidine tag and SpeI/NotI restriction sites. Mutations were introduced using the QuikChange method (Stratagene), and the PCR product was subcloned into pFastBac1 for protein expression in Sf9 cells. For focus formation assays, full-length ALK variants were subcloned into the pcDNA3.1(-) vector (Invitrogen).

B. Recombinant protein expression and purification. Spodoptera frugiperda Sf9 cells at $1.5 - 2 \times 10^6$ /ml were infected with recombinant baculovirus, and harvested by centrifugation after 3 days. Cells expressing histidine-tagged ALK variants were lysed by sonication in 100ml of lysis buffer (50 mM sodium phosphate buffer, pH 8.0, 300 mM NaCl, 10mM imidazole, 4mM β-mercaptoethanol, and protease inhibitor cocktail (Roche)). After centrifugation at 40,000 x g for 30 minutes to remove insoluble cell debris, the supernatant was incubated with Ni-NTA agarose beads (Qiagen) for 1 hour at 4° C. The Ni-NTA beads were washed with 50 column volumes of lysis buffer, pH 8.0 containing 20mM imidazole and bound ALK TKD protein was eluted in lysis buffer, pH 8.0 containing 300mM imidazole. Eluted ALK TKD protein was incubated with 1μM YopH phosphatase for 12 h at 4°C to reverse autophosphorylation that occurred during expression. YopH was then removed using a cation exchange column after reducing NaCl concentration to 100 mM. Eluted protein was then further purified using a butyl sepharose HP column (GE Healthcare) equilibrated with 1M $(NH_4)_2SO_4$), 25mM HEPES pH 7.0, 150 mM NaCl and 2mM DTT (HIC buffer A), eluting with a 20 column volume linear gradient to 0 M $(NH_4)_2SO_4$) , 25mM HEPES pH 7.0, 150 mM NaCl and 2mM DTT (HIC buffer B). ALK TKD protein was finally subjected to a size exclusion chromatography step using a Superdex 200 column (GE Healthcare) equilibrated in 25 mM HEPES pH 7.4, 150 mM NaCl, 4 mM DTT.

C. Peptide phosphorylation assays. In vitro kinase assays measuring $\gamma-^{32}$P incorporation into peptide substrate were performed as described (1). The substrate was a peptide mimic of the ALK activation loop with sequence: biotin- ARDIYRASYYRKG-GCAMLPVK (CanPeptide). Enzyme concentrations for unphosphorylated ALK-TKD variants were fixed at 50nM. Under these assay conditions, reaction rates were linear with respect to enzyme concentration and time. Assays were performed in 100 mM HEPES pH 7.4, 150 mM NaCl, 2mM DTT, 10 mM $MgCl_2$ and 0.5 mg/ml BSA at 25°C. k_{cat} values were measured by varying peptide concentration from 0.015625 mM to 2mM at excess ATP (2mM). Samples were taken at each time point, spotted onto P81 Ion Exchange Cellulose Chromatography paper (Reaction Biology Corp) and quenched with 0.5 % phosphoric acid. Liquid scintillation was used to measure incorporated radioactivity on each paper filter. Initial rates were all determined from the linear portion of the enzyme reaction (when substrate depletion or product formation is < 10 %), normalized for enzyme concentration, and fit to the Michaelis-Menten equation (vo = vmax[S]/(Km+[S])) using GraphPad Prism 5.0.

D. Focus formation assays. For focus-formation assays, full-length ALK variants were subcloned into the pcDNA3.1 (-) (Invitrogen). Low-passage (typically < 15) NIH 3T3 cells at approximately 60 % to 70 % confluence were transfected with full-length mutated ALK constructs using Lipofectamine 2000 (Invitrogen). After 2 days of recovery, transfected NIH3T3 cells were divided into 2 groups and plated. The first (focus formation) group was plated on 10-cm dishes and left to reach full confluence and to form foci in DMEM with GlutaMaX with 5 % calf serum. For the second (colony formation) group, serially diluted cells were plated in wells of 6 plates in DMEM with GlutaMax with 10 % calf serum and 0.5 mg/ml G418 to select colonies for counting.

Medium was then changed every 3 days until foci and colonies were formed, which typically takes 2-3 weeks. Cells were then fixed in 3.7 % formaldehyde in phosphate-buffered saline (PBS) for 5 minutes and then stained with 0.05 % crystal violet in distilled water for 30 minutes. Data are reported as a transformation index, which is the number of foci corrected for transfection efficiency (estimated by the count of G418-resistant colonies), and normalized to a parallel assessment of transformation by the activating F1174L mutation, which was given the arbitrary transformation index of 1.0. Each independent experiment was performed in triplicate, and mean values are reported with standard deviation.

2. Computational Methods

A. Molecular Dynamics Simulations. Simulations and analysis were carried out using the BioPhysCode software suite (2). The initial structures of ALK were as previously reported (3). All homology models were constructed using MODELLER (4) and all mutations were introduced using a BioPhysCode Automacs routine based on MODELLER. The inactive wild-type ALK TKD structure (residues 1096-1399) was taken from PDB entry 3LCS (5). Missing residues 1084-1095 and 1400-1405 were added to the model based on PDB entry 4FNW using MODELLER. Mutated structures were generated using MODELLER by making point mutations to the modified inactive wild-type model. A homology model of active ALK TKD was generated with MODELLER, using as the primary template the active insulin receptor TKD structure (PDB entry 1IR3), with which ALK TKD shares 46% sequence identity. Residues 1097-1399 were modeled from 3LCS, whereas residues 1084-1096 and 1400-1405 were again modeled from 4FNW. All structures were modeled without bound substrate. Simulations were run with Automacs using GROMACS (6) with the CHARMM27 force field (7) with TIP3P explicit solvent (8) in a periodic water box with at least 12 Åbetween the protein and box edge. An ionic concentration of 0.15 M NaCl was used and the final charge of the full system was zero. Minimization was carried out using steepest descent and the system was equilibrated first at constant volume, then at constant pressure using Berendsen thermostat (9) before production MD simulations were carried out constant pressure

using Parinello-Rahman (10). Equilibration and production MD runs were carried out at constant temperature using velocity rescaling (11), with linear center of mass motion removal. LINCS (12) was used to constrain all bonds during equilibration and hydrogen bonds were constrained during production MD. Particle mesh Ewald electrostatics (3) was used to account for long-ranged interactions (13). Simulations were run for a total of 101 ns and two replicates were performed for each simulation.

B. Molecular Dynamics Analysis. Analysis was performed, unless otherwise noted, on the last 100 ns and the two replicates were averaged together. Structures were sampled from each trajectory at 20 ps intervals, resulting in a total of 5001 structures for analysis. Plotting was performed with Omnicalc using matplotlib (14).

B.1. Hydrogen bond occupancy. Each amino acid is considered to have a maximum of 3 possible hydrogen bonds: a main chain donor, a main chain acceptor, and the side chain — meaning that some residues such as Arg or Asp can have more than one side chain hydrogen bond in a single frame; however, bonds are counted uniquely so that this could only happen if e.g. Arg-i and Asp-j side chains make both possible hydrogen bonds. For each structure in a trajectory and for each bond (see Fig. S1) the hydrogen bond occupancy ($\mathbf{O}$) was calculated by dividing the number of frames with a hydrogen bond is observed by the total number of frames. After computing the occupancy for each residue i in the inactive WT ($\mathbf{O_{WT,i}}$) and residue i in the inactive mutant ($\mathbf{O_{MUT,i}}$) the occupancy difference in the mutant MUT for residue i ($\mathbf{\Delta_{MUT,i}}$) was calculated as $\mathbf{\Delta_{MUT,i}} = \mathbf{O_{MUT,i}} - \mathbf{O_{WT,i}}$. For each residue i, if $\mathbf{\Delta_{MUT,i}} > threshold$, then $\mathbf{\Delta_{MUT,i}}$ is added to an accumulator ($\mathbf{\Delta_{MUT,Total}}$). Here $threshold$ is set to 0.75 and a mutation considered to have a different occupancy than WT if $\mathbf{\Delta_{MUT,Total}}$ is nonzero and exceeds a second threshold. This threshold is varied to get a sensitivity of the chosen value to the prediction accuracy as discussed in the main text.

B.2. RMSD analysis. RMSD was computed for αC helix and activation loop residues after aligning the rest of the C_α carbon atoms with the active and the inactive reference states. The temporal RMSD plots were generated for each mutant system as depicted in the example plots in Fig. S2. A threshold standard deviation of RMSD change of > 2 Å either in the αC helix or the activation loop in a given system was scored as activating. The resultant analysis yielded similar BACC to that obtained using hydrogen bond occupancy. Since the hydrogen bond occupancy can yield residue level information, we utilize that method in the final analysis and comparison to experiments.

C. Metadynamics. Well tempered metadynamics (15) (WTMD) was used to sample the large scale configurational space between the inactive and the active configurations of ALK to assess conformational the conformational changes in the kinase subdomains that provide distinguishable features (mainly the activation loop and the αC helix), with the active configuration having α C-in and the activation loop being extended, and the inactive active configuration of ALK having α C-out and the activation loop not being extended: inactive configuration of ALK (3). The biased simulations were performed using PLUMED 2.3.5 patched with GROMACS 5.0.7. Metadynamics accelerates rare events along certain collective variables (CVs) that are functions of positional coordinates. Here we use a simple geometry based CV or path based CV: RMSD to the active structure of ALK as CV1 and RMSD to inactive structure of ALK as CV2. The RMSD is calculated based on only the non-translational and non-rotational motion of the alpha carbon atoms in the protein backbone. Metadynamics involves adding an external history dependent Gaussian potential for the system to be able to cross the barrier.

$$V(\vec{s},t) = \sum_{k\tau<t} W(k\tau) exp\left(-\sum_{i=1}^{d} \frac{(s_i - \bar{s}_i(k\tau))^2}{2\sigma_i^2} \right) \quad [1]$$

To parallelize the metadynamics calculations, we use four walkers meaning four parallel simulations that sample the configurational space simultaneously and are cognizant of the deposited Gaussian potentials by other walkers on the CV grid space through a file sharing system of PLUMED. Here in eq 1, σ_i is the width of the Gaussian for the CV s_i, W represents the height of the Gaussian and τ is the Gaussian deposition stride. WTMD uses decaying Gaussian height to ensure smoother convergence.

$$W(k\tau) = W_o exp\left(-\frac{V(\vec{s},k\tau)}{k_B \Delta T} \right) \quad [2]$$

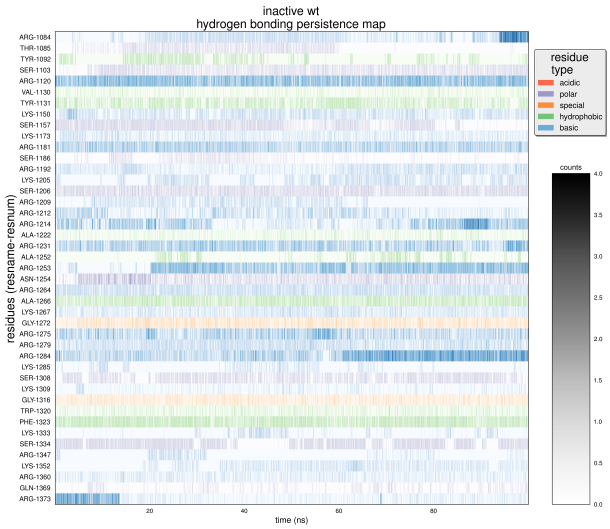
In the larger time limits, the free energy for the CV space is obtained as,

$$V(S, t\to\infty) = -\frac{\Delta T}{T+\Delta T} F(S) + C \quad [3]$$

We set in the PLUMED script, the energy in kcal/mol and length in Å. The parameters used in this study to perform WTMD are:bias factor $\gamma = \frac{T+\Delta T}{T} = 20$, height $= 0.6$ and pace $= 500$.

C.1. Convergence of free energy profile obtained through metadynamics. The CVs that we have employed, CV1:RMSD to active and CV2: RMSD to inactive span a very large configurational space, but is required to capture the transition between the active and the inactive states. Initially, we utilized the configurations that resulted from equilibration of the structures from PDB/homology modeling as the inactive and active reference structures for CV1 and 2. After a first round of metadynamics, we identified the zones corresponding to the inactive and active states and then utilized representative structures from these

ALK WT inactive Hbonds contact map



ALK WT inactive replicate 400 ns Hbonds contact map

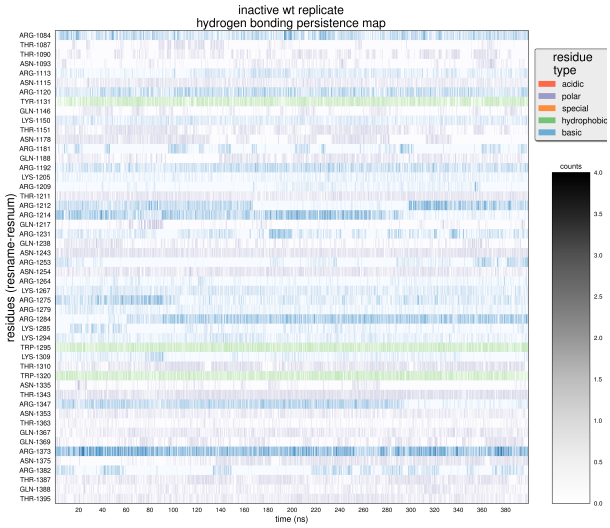


Fig. S1. Inactive hydrogen bond contact maps. Residues are colored by type of side chain. Darkness is determined by the number of hydrogen bonds a residue participates in during a single frame. Backbone and side chain contributions are taken together. Only residues that participate in at least one hydrogen bond for at least 10% of frames are shown.

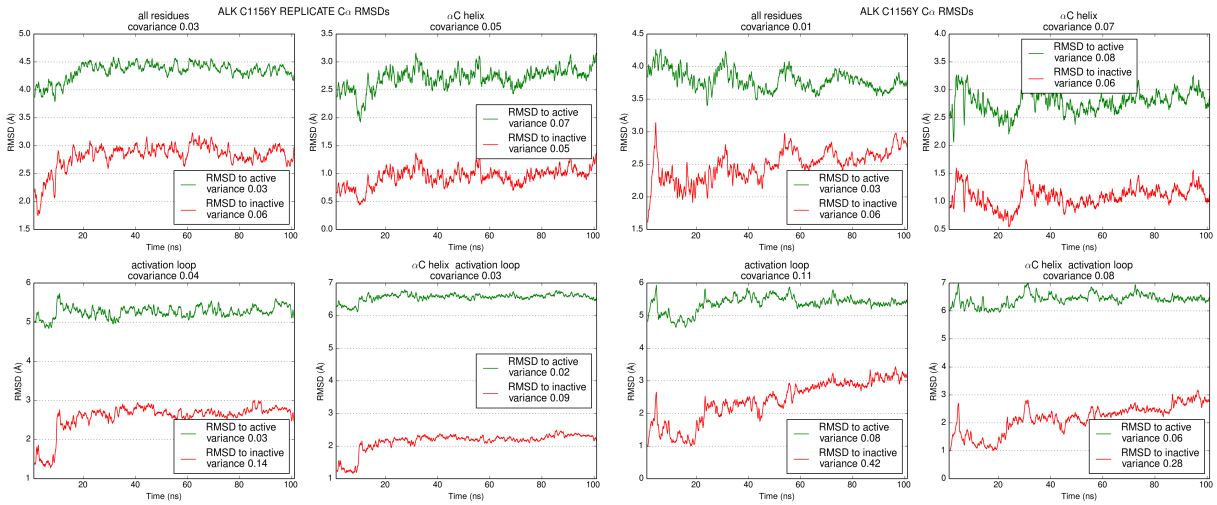


Fig. S2. RMSD evolution of the C1156Y mutant as an example in two (replicate) trajectories. Overall RMSD as well as those of αC helix and the activation loop regions are shown.

zones in the definition of the CVs. This one-step iterative procedure led to superior convergence of the landscapes. The choice of the CVs and the iterative procedure culminates in a very long biased simulation time. Hence, we simulate to ensure the convergence of certain zones of interest in the free energy landscape to with less than 0.5 kcal/mol for 100 ns based on evolution of the free energy of these zones or states according to,

$$F_s = -k_B T log\left(\int\int e^{-\beta \hat{F}(s_1,s_2)} ds_1 ds_2\right), \quad [4]$$

where F_s is the free energy of the state, $\hat{F}(s_1, s_2)$ is the free energy of the value at that collective variable coordinates (s_1, s_2). These zones represent the transition from the inactive-like to active-like configurations of ALK as depicted in the main text. The nature of metadynamics is such that it discourages the system from visiting more frequently visited regions. However, a convergence analysis needs to be performed to ensure reproducibility. We perform such an analysis in Fig. S3.

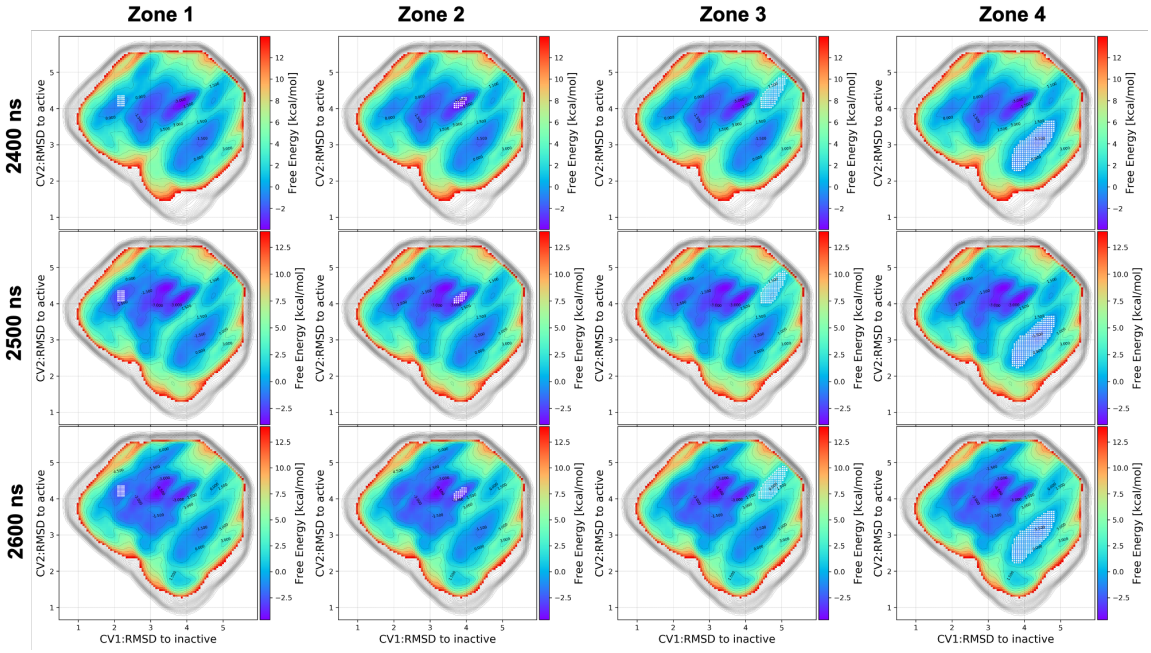


Fig. S3. We compute and depict the evolution of the quantity F_{z1}-F_{z4} or free energy for zones 1-4 (columns) using eq 4. The rows represent the free energy values after 2400 (top), 2500 (middle), and 2600 (bottom) ns of metadynamics, respectively. The converged values of the free energies are tabulated in the main text.

A summary of the four zones is depicted in Fig. S4. The zones are identified as regions in the free energy landscape within $5k_BT$ of the minimum free energy value in that zone.

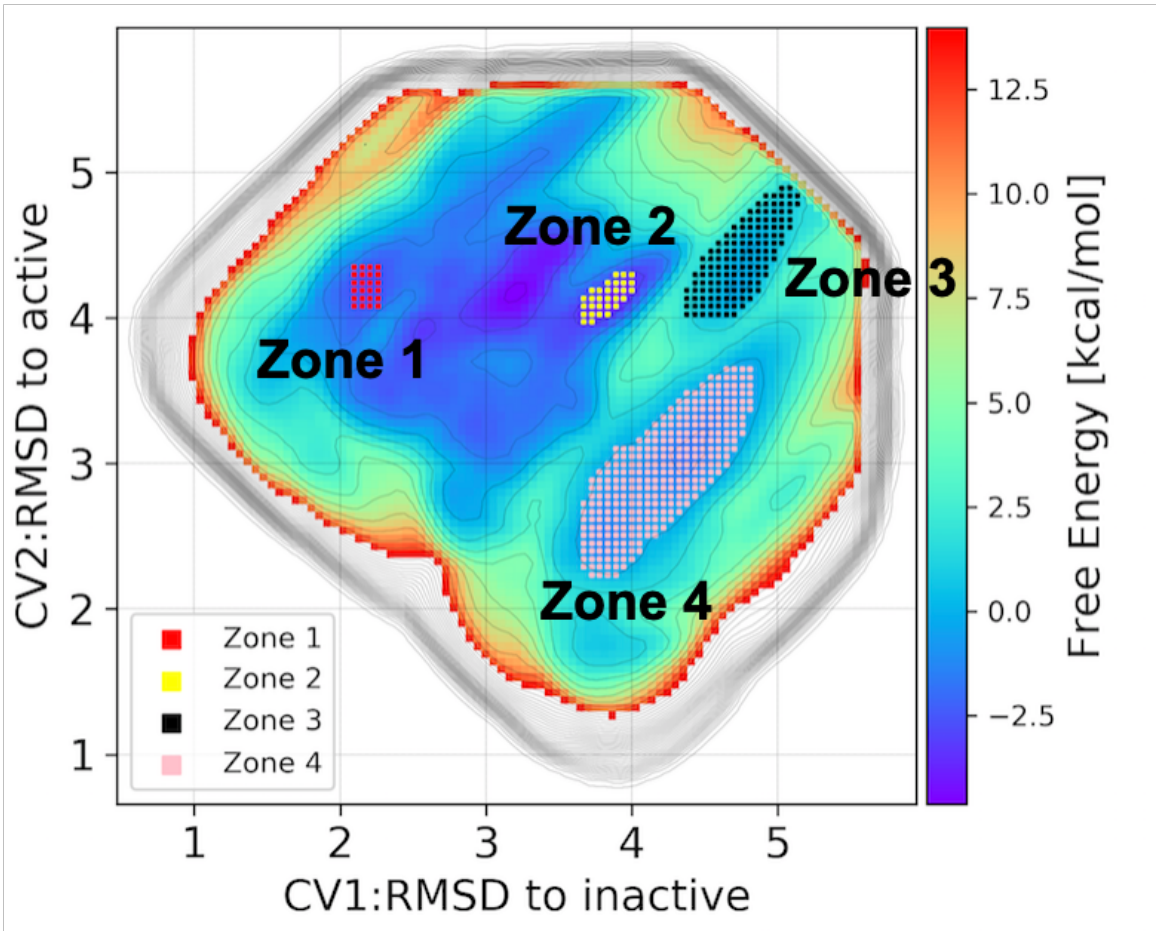


Fig. S4. Four converged zones identified on the free energy landscape subjected to further analysis.

D. Machine Learning Algorithm.

D.1. Curation of Dataset. A pan-kinase mutation dataset was constructed via text mining of the UniProt database using a Perl script. The resulting data set was validated by searching the literature for a subset of the entire dataset to ensure that class assignments were correct. The final set used in this work contained 829 total point mutations, with 230 positive, activating mutations, and 599 negative, non-activating mutations. For each mutation, a feature vector with 59 elements was generated, addressing chemical properties of the wild type and mutant residues.

D.2. Construction of Feature Vectors. For each mutation, a feature vector of the following values was constructed. This leads to a feature vector for each mutation with 59 elements. Each element of the resulting vectors is normalized so that all values are in [-1,1]. A large number of the elements will be zero for each mutation. The following is a list of all the features:

1. Wild type residue (one feature element for each of the 20 amino acids)
2. Mutated residue (one feature element for each of the 20 amino acids)
3. Wild type residue type (from aliphatic, acidic, basic, aromatic, and polar)
4. Mutated residue type (from aliphatic, acidic, basic, aromatic, and polar)
5. Difference between wild type and mutated residue for following:
   (a) Kyte-Doolittle hydropathy
   (b) Free energy of solvation
   (c) Normalized van der Waals radius
   (d) Polarity difference
   (e) Charge difference

6. Whether the mutation falls in one of the following kinase subdomains:
   (a) nucleotide binding loop
   (b) αC helix
   (c) catalytic loop
   (d) activation loop

D.3. Construction of Data Matrix and Training. Feature vectors were generated for each of the 829 mutations via a python script that extracted features from the data. The data file has the following information for each kinase.

1. The name of the kinase (BRAF, ALK, etc.)
2. The wild type residue (before point mutation)
3. The mutated residue (after point mutation)
4. The location of the point mutation (residue number)
5. Label (+1: activating, -1: non-activating)

A data matrix was generated with the features (normalized numerical values) and labels for each mutation. Data were divided into a training set and a test set. The test set consists of all 41 ALK mutations in Fig. 4A. The training set consists of all of the other 784 mutations. Since the data were imbalanced, the SMOTE algorithm, for up sampling of the minority class, was performed on the training set so that it consisted of an equal number of activating and non-activating mutations. SMOTE was only applied on the training set to prevent overfitting (16–18). 5-fold cross validation was then performed on the training set to determine the optimal hyperparameters that maximized both the f1 score and the ROC AUC score. The following algorithms were used in this study: Support Vector Machine (SVM), Logistic Regression, Neural Net, and Random Forest. The data were utilized to determine the optimal hyperparameters. Namely, the hyperparameters were tuned in cross-validation. After training the models using the training data and optimized hyperparameters, the model was evaluated on the test data.

D.4. Support Vector Machine (SVM). Model Choice: The Support Vector Machine (SVM), with the Radial Basis Function (RBF) kernel, was chosen since the data were numerical and the number of samples was much greater than the dimensions of the feature space. Model Training and Hyperparameter Search: The training data were utilized to determine the optimal hyperparameters. SVM has a number of parameters that can be optimized. For the SVM RBF, the error penalty, C, and the Gaussian width γ can be optimized. The error penalty C controls how smooth the decision surface is, with larger values of C leading to an increasingly jagged boundary that attempts to classify every example correctly. The Gaussian width γ controls how large of a region in feature space (or any mapping of feature space) that the training examples take up, with larger values meaning training examples are 'felt' in a smaller region. Both C and γ can be tuned in cross-validation (19). To this end, a grid search was implemented over all combinations of values of $\gamma \in [1 \times 10^{-5}, 1 \times 10^{4}]$ increasing by a factor of 10 in each iteration, $C \in \{0.01, 0.1, 1, 2, 3, 4, 5\}$ for loss functions that maximize one of F1, ROC AUC. The F1 score is a weighted average of the precision and recall, both of which are defined later when discussing the measures used in evaluating the performance of the model. The F1 score reaches its best value at 1 and worst score at 0. The ROC is a plot of the false positive rate (x-axis) versus the true positive rate (y-axis) for a number of different candidate threshold values between 0.0 and 1.0. ROC AUC calculates the area under the ROC curve; the best possible AUC is 1 while the worst is 0.5 (the 45 degrees random line). F1 and ROC AUC are more representative loss functions, than accuracy, since the data are imbalanced.

The grid search was conducted by performing 5-fold cross validation. The training data set was shuffled randomly and then split into 5 groups. For each unique group, that group was taken as the test data set and the remaining groups as a training data set. A model was fit on the training set and evaluated on the test set. The model was discarded after retaining the f1 and ROC AUC scores and the process was repeated for each unique group. The skill of the model with that particular combination of hyperparameter values was then summarized using the sample of model evaluation scores. For the training set used here, $C = 5, \gamma = 0.1$, kernel=rbf were found to be the hyperparameters that maximized the ROC AUC and F1 loss functions. A plot of the ROC AUC scores for combinations of Gamma and C values tested during cross validation is depicted in Fig. S5. In addition, we performed a statistical test (F-test) to determine the relevance of the features in the ML algorithm S6. The F-test is a statistical test used to compare between models and to check if the difference in performance is significant. In this case, iterations of hypothesis testing are done where one model, X, contains "n" features, and model Y has "n+1" features. The least squares errors in both models are compared and a p-value is calculated to determine whether the difference in errors between model X and Y are significant or introduced by chance. A significant difference means that the feature added provided a meaningful contribution to the performance and improvement of the ML algorithm.

D.5. Neural Network. Model Choice: The neural net was constructed using the Keras framework. The optimizer was chosen to be "stochastic gradient descent" and the loss function was designated as "binary cross entropy." Given the size of the data set and the number of features at our disposal, a 3 layered neural net was constructed. The first layer has 8 units and an input dimension of 59; the second layer has 8 units and the "tanh" activation function; the third layer has 1 unit and the "sigmoid" activation function. The max epoch number was set to 500 and the batch size was set to 32 (20).

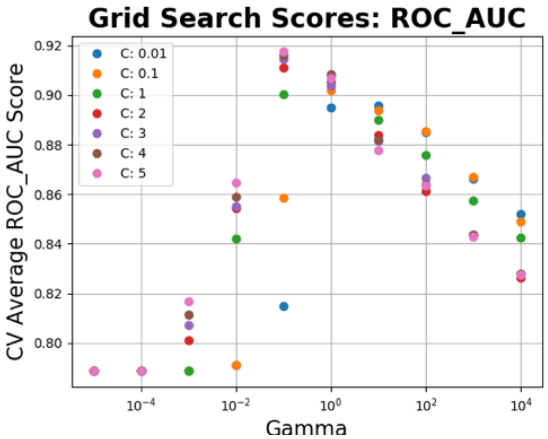
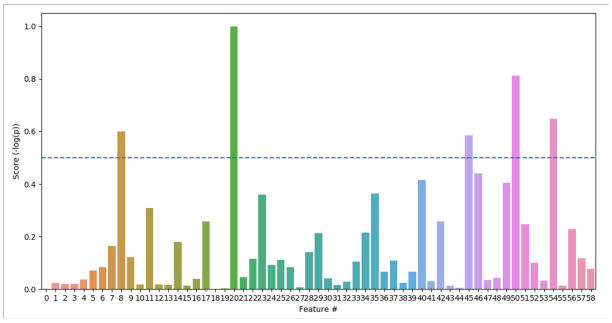


Fig. S5. Plot of ROC AUC scores for combinations of γ and C values tested during cross validation. Optimal combination of parameters found to be: $C = 5$, $\gamma = 0.1$, 'kernel'='rbf'.



Feature #	Feature
8	WT K
20	MUTANT A
45	MUTANT aliphatic
50	Kyte-Doolittle hydropathy difference
54	Polarity difference

Fig. S6. (top) F-Test Scores for features in SVM Algorithm. Dotted Line drawn at 0.5 to highlight most important features. (bottom) Features with greatest scores from F-test analysis

D.6. Logistic Regression. Model Training and Hyperparameter Search: The training data was utilized to determine the optimal hyperparameters. The following were the parameters that can be optimized for Logistic Regression: solver, algorithm to use in the optimization problem; C, inverse of regularization strength; and, multi-class, where if 'ovr' is chosen, a binary problem is fit for each label and if 'multinomial' is chosen, the loss minimized is the multinomial loss fit across the entire probability distribution, even when the data is binary. All of these parameters can be tuned through cross-validation (21). To this end, a grid search was implemented over all combinations of values of solver $\in \{0.01, 0.1, 1, 2, 3, 4, 5\}$, multi-class $\in$ ['ovr','multinomial'], and $C \in \{0.001, 0.01, 0.1, 1, 10, 100, 1000\}$. For the training set used here, 'solver' = 'newton-cg', 'multi-class' = 'multinomial', 'C' = 100 were found to be the hyperparameters that maximized the ROC AUC and F1 loss functions.

D.7. Random Forest. Model Training and Hyperparameter Search: The training data was utilized to determine the optimal hyperparameters. The following were the parameters that can be optimized for Random Forest: n-estimators, the number of trees in the random forest; max-features, the number of features to consider at every split; max-depth, the maximum number of levels in each tree; min-samples-split, the minimum number of samples required to split a node; min-samples-leaf, the minimum number of samples required at each leaf node; and, bootstrap, the method of selecting samples for training each tree. All of these parameters can be tuned through cross-validation (22). Since this is a large hyperparameter space, a randomized grid search was implemented. This means that not all parameter values are tried out, but rather a fixed number of parameter settings is sampled from the specified distributions. This was done over all combinations of values of n-estimators $\in [200, 2000]$ increasing by 180 in each iteration, max-features $\in$ ['auto','sqrt'], max-depth $\in [10, 110] \cup$ ['None'] increasing by 9 each iteration, min-samples-split $\in [2, 5, 10]$, min-samples-leaf $\in [1, 2, 4]$, and boostrap $\in [True, False]$ for loss functions which maximize one of [F1, ROC AUC]. For the training set used here, [n-estimators=1200,min-samples-split=10,min-samples-leaf=1, max-features="sqrt",max-depth=None,bootstrap=True] were found to be the hyperparameters that maximized the ROC AUC and F1 loss functions.

D.8. Evaluation. After training the models using the training data and optimized hyperparameters, the model can be evaluated on the test data. Using the trained model, we made predictions on the labels (1: activating, -1: non-activating) of the test set. The following measures are usually used in evaluating the performance of the model: TP = # of true positives, TN = # of true negatives, FP = # of false positives, and FN = # of false negatives, where BACC = Balanced Accuracy, TPR = True Positive Rate, TNR = True Negative Rate, TN = True Negative Rate, FPR = False Positive Rate, TPR = True Positive Rate, TNR= True Negative Rate, and FNR = False Negative Rate. All ML implementations were executed in Python. Here: TPR=TP/(TP+FN), FPR=FP/(FP+TN), FNR=1-TPR, TNR=1-FPR, BACC=(TPR+TNR)/2, and Accuracy=(TP+TN)/(TP+TN+FP+FN).

E. GitHub Repository. The machine learning models, training data, and documentation are available for download from https: //github.com/kksuresh25/Cancer-AI

The molecular dynamics and metadynamics files are available for download from https: //github.com/KesPatil/ALK_files_2021_PNAS.git

3. Additional SI Figures and Tables

ALK	H-bond donor	H-bond acceptor	# simulation bond is labile
	Arg-1181 side	Glu-1197 side	47
	Arg-1231 side	Glu-1384 side	40
	Arg-1253 side	Asp-1249 side	33
	Arg-1284 side	Asp-1163 side	30
	Arg-1279 side	Asp-1163 side	23
	Arg-1275 side	Asp-1276 side	22
	Arg-1284 side	Asp-1276 side	21
	Listed bonds/total labile bonds		216/390

Table S1. Labile hydrogen bonds: all listed H-bonds are between residue side chains (side) but main chains were also considered.

References

1. Bresler SC, et al. (2011) Differential inhibitor sensitivity of anaplastic lymphoma kinase variants found in neuroblastoma. *Sci Transl Med* 3(108):108ra114.
2. (2020) Biophyscode. https://biophyscode.github.io.
3. Bresler SC, et al. (2014) Alk mutations confer differential oncogenic activation and sensitivity to alk inhibition therapy in neuroblastoma. *Cancer cell* 26(5):682–694.
4. Sali A, Blundell T (1994) Comparative protein modelling by satisfaction of spatial restraints. *Protein structure by distance analysis* 234(3):779–815.
5. Lee CC, et al. (2010) Crystal structure of the ALK (anaplastic lymphoma kinase) catalytic domain. *Biochemical Journal* 437:425–437.
6. Páll S, Kutzner C, Abraham MJ, Hess B, Lindahl E (2015) Tackling Exascale Software Challenges in Molecular Dynamics Simulations with GROMACS. *Proc. of EASC 2015 LNCS* 8579:3–27.
7. MacKerell AD, et al. (1998) All-Atom Empirical Potential for Molecular Modeling and Dynamics Studies of Proteins [†]. *The Journal of Physical Chemistry B* 102(18):3586–3616.
8. Jorgensen WL, Chandrasekhar J, Madura JD, Impey RW, Klein ML (1983) Comparison of simple potential functions for simulating liquid water. *The Journal of Chemical Physics* 79(2):926–935.
9. Berendsen HJC, Postma JPM, van Gunsteren WF, DiNola A, Haak JR (1984) Molecular dynamics with coupling to an external bath. *The Journal of Chemical Physics* 81(8):3684–3690.
10. Parrinello M, Rahman A (1981) Polymorphic transitions in single crystals: A new molecular dynamics method. *Journal of Applied Physics* 52(12):7182–7190.
11. Bussi G, Donadio D, Parrinello M (2007) Canonical sampling through velocity rescaling. *The Journal of Chemical Physics* 126(1):14101.
12. Hess B, Bekker H, Berendsen HJC, Fraaije JGEM (1997) Lincs: A linear constraint solver for molecular simulations. *Journal of Computational Chemistry* 18(12):1463–1472.
13. Essmann U, et al. (1995) A smooth particle mesh Ewald method. *The Journal of Chemical Physics* 103(19):8577–8593.
14. Hunter JD (2007) Matplotlib: A 2d graphics environment. *Computing In Science & Engineering* 9(3):90–95.
15. Barducci A, Bussi G, Parrinello M (2008) Well-tempered metadynamics: a smoothly converging and tunable free-energy method. *Physical Review Letters* 100(2):020603.
16. Chawla NV, Bowyer KW, Hall LO, Kegelmeyer WP (2002) Smote: Synthetic minority over-sampling technique. *J. Artificial Intelligence Research* 16:321–357.
17. Fernandez, A. GSHFCNV (2018) Smote for learning from imbalanced data: Progress and challenges, marking the 15-year anniversary. *Journal of Artificial Intelligence Research* 61:863–905.

18. Wang J, Xu M, Wang H, Zhang J (year?) Classification of imbalanced data by using the smote algorithm and locally linear embedding in *ICSP2006 Proceedings*.
19. (2020) Support vector machines. https://scikit-learn.org/; module= svm.
20. (2020) Keras: The python deep learning library. https://keras.io/.
21. (2020) Sklearn linear model logistic regression. https://scikit-learn.org/; module= sklearn linear model Logistic Regression.
22. (2020) sklearn ensemble random forest classifier. https://scikit-learn.org/; module= sklearn ensemble Random Forest Classifier.