ARTICLE TYPE

Prioritizing 2nd and 3rd order interactions via support vector ranking using sensitivity indices on time series Wnt measurements - Part B† [work in progress]

shriprakash sinha*◕, [Author list to be updated]

It is widely known that the sensitivity analysis plays a major role in computing the strength of the influence of involved factors in any phenomena under investigation. When applied to expression profiles of various intra/extracellular factors that form an integral part of a signaling pathway, the variance and density based analysis yields a range of sensitivity indices for individual as well as various combinations of factors. These combinations denote the higher order interactions among the involved factors. Computation of higher order interactions is often time consuming but they give a chance to explore the various combinations that might be of interest in the working mechanism of the pathway. For example, in a range of fourth order combinations among the various factors of the Wnt pathway, it would be easy to assess the influence of the destruction complex formed by APC, AXIN, CSKI and GSK3 interaction. But the effect of these combinations vary over time as measurements of fold changes and deviations in fold changes vary. In this work, after estimating the individual effects of factors for a higher order combination, the individual indices are considered as discriminative features. A combination, then is a multivariate feature set in higher order (>2). With an excessively large number of factors involved in the pathway, it is difficult to search for important combinations in a wide search space over different orders. Exploiting the analogy with the issues of prioritizing webpages using ranking algorithms, for a particular order, a full set of combinations of interactions can then be prioritized based on these features using a powerful ranking algorithm via support vectors. Recording the changing rankings of the combinations over time reveals of how higher order interactions behave within the pathway and when an intervention might be necessary to influence the interaction within the pathway. This could lead to development of time based therapeutic interventions.Code has been made available on Google drive at `https://drive.google.com/folderview?id=0B7Kkv8wlhPU-V1Fkd1dMSTd5ak0&usp=sharing`

1 Introduction

1.1 A short review

Sharma[1]'s accidental discovery of the Wingless played a pioneering role in the emergence of a widely expanding research field of the Wnt signaling pathway. A majority of the work has focused on issues related to • the discovery of genetic and epigenetic factors affecting the pathway (Thorstensen *et al.*[2] & Baron and Kneissel[3]), • implications of mutations in the pathway and its dominant role on cancer and other diseases (Clevers[4]), • investigation into the pathway's contribution towards embryo development (Sokol[5]), homeostasis (Pinto *et al.*[6], Zhong *et al.*[7]) and apoptosis (Pećina-Šlaus[8]) and • safety and feasibility of drug design for the Wnt pathway (Kahn[9], Garber[10], Voronkov and Krauss[11], Blagodatski *et al.*[12] & Curtin and Lorenzi[13]). Approximately forty years after the discovery, important strides have been made in the research work involving several wet lab experiments and cancer clinical trials (Kahn[9], Curtin and Lorenzi[13]) which

* *Corresponding Author : shriprakash sinha.*
◕ *Author is a buddhist monk and currently working as an independent researcher. Address - 104-Madhurisha Heights Phase 1, Risali, Bhilai - 490006, INDIA; E-mail : sinha.shriprakash@yandex.com*
† Electronic Supplementary Information (ESI) available: [details of any supplementary information available should be included here]. See DOI: 10.1039/b000000x/

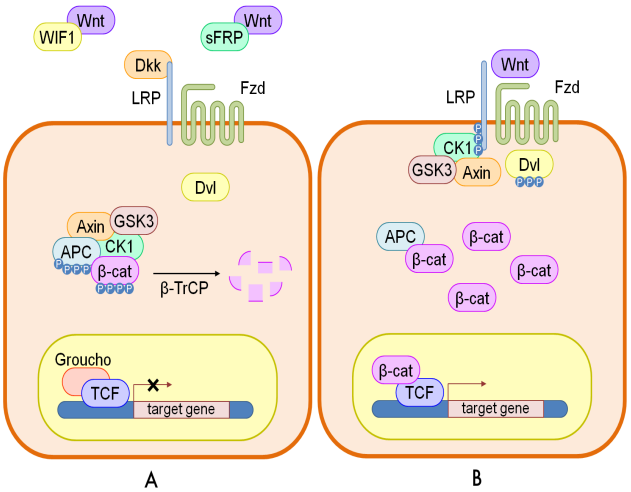

Fig. 1 A cartoon of Wnt signaling pathway contributed by Verhaegh *et al.*[18]. Part (A) represents the destruction of *β-catenin* leading to the inactivation of the Wnt target gene. Part (B) represents activation of Wnt target gene.

have been augmented by the recent developments in the various advanced computational modeling techniques of the pathway. More recent informative reviews have touched on various issues related to the different types of the Wnt signaling pathway and have stressed not only the activation of the Wnt signaling pathway via the Wnt proteins (Rao and Kühl[14]) but also the on the secretion mechanism that plays a major role in the initiation of the Wnt activity as a prelude (Yu and Virshup[15]).

The work in this paper investigates some of the current aspects of research regarding the pathway via sensitivity analysis while using static (Jiang *et al.*[16]) and time series (Gujral and MacBeath[17]) gene expression data retrieved from colorectal cancer samples.

1.2 Canonical Wnt signaling pathway

Before delving into the problem statement, a brief introduction to the Wnt pathway is given here. From the recent work of Sinha[19], the canonical Wnt signaling pathway is a transduction mechanism that contributes to embryo development and controls homeostatic self renewal in several tissues (Clevers[4]). Somatic mutations in the pathway are known to be associated with cancer in different parts of the human body. Prominent among them is the colorectal cancer case (Gregorieff and Clevers[20]). In a succinct overview, the Wnt signaling pathway works when the Wnt ligand gets attached to the Frizzled(FZD)/LRP coreceptor complex. FZD may interact with the Dishevelled (DVL) causing phosphorylation. It is also thought that Wnts cause phosphorylation of the LRP via casein kinase 1 ($CK1$) and kinase $GSK3$. These developments further lead to attraction of Axin which causes inhibition of the formation of the degradation complex. The degradation complex constitutes of $AXIN$, the β-$catenin$ transportation complex APC, $CK1$ and $GSK3$. When the pathway is active the dissolution of the degradation complex leads to stabilization in the concentration of β-$catenin$ in the cytoplasm. As β-$catenin$ enters into the nucleus it displaces the $GROUCHO$ and binds with transcription cell factor TCF thus instigating transcription of Wnt target genes. $GROUCHO$ acts as lock on TCF and prevents the transcription of target genes which may induce cancer. In cases when the Wnt ligands are not captured by the coreceptor at the cell membrane, $AXIN$ helps in formation of the degradation complex. The degradation complex phosphorylates β-$catenin$ which is then recognized by $FBOX/WD$ repeat protein β-$TRCP$. β-$TRCP$ is a component of ubiquitin ligase complex that helps in ubiquitination of β-$catenin$ thus marking it for degradation via the proteasome. Cartoons depicting the phenomena of Wnt being inactive and active are shown in figures 1(A) and 1(B), respectively.

2 Problem statement & sensitivity analysis

It is widely known that the sensitivity analysis plays a major role in computing the strength of the influence of involved factors in any phenomena under investigation. When applied to expression profiles of various intra/extracellular factors that form an integral part of a signaling pathway, the variance and density based analysis yields a range of sensitivity indices for individual as well as various combinations of factors. These combinations denote the higher order interactions among the involved factors. Computation of higher order interactions is often time consuming but they give a chance to explore the various combinations that might be of interest in the working mechanism of the pathway. For example, in a range of fourth order combinations among the various factors of the Wnt pathway, it would be easy to assess the influence of the destruction complex formed by APC, AXIN, CSKI and GSK3 interaction. But the effect of these combinations vary over time as measurements of fold changes and deviations in fold changes vary. So it is imperative to know how an interaction or a combination of the involved factors behave in time and a procedure needs to be developed that can track the behaviour by exploiting the influences of the involved factors.

In this work, after estimating the individual effects of factors for a higher order combination, the individual indices are considered as discriminative features. A combination, then is a multivariate feature set in higher order (>2). With an excessively large number of factors involved in the pathway, it is difficult to search for important combinations in a wide search space over different orders. Exploiting the analogy with the issues of prioritizing webpages using ranking algorithms, for a particular order,

a full set of combinations of interactions can then be prioritized based on these features using a powerful ranking algorithm via support vectors (Joachims[80]). Recording the changing rankings of the combinations over time reveals of how higher order interactions behave within the pathway and when an intervention might be necessary to influence the interaction within the pathway. This could lead to development of time based therapeutic interventions.

2.1 Sensitivity analysis

Seminal work by Russian mathematician Sobol'[21] lead to development as well as employment of SA methods to study various complex systems where it was tough to measure the contribution of various input parameters in the behaviour of the output. A recent unpublished review on the global SA methods by Iooss and Lemaître[22] categorically delineates these methods with the following functionality • screening for sorting influential measures (Morris[23] method, Group screening in Moon *et al.*[24] & Dean and Lewis[25], Iterated factorial design in Andres and Hajas[26], Sequential bifurcation in Bettonvil and Kleijnen[27] and Cotter[28] design), • quantitative indicies for measuring the importance of contributing input factors in linear models (Christensen[29], Saltelli *et al.*[30], Helton and Davis[31] and McKay *et al.*[32]) and nonlinear models (Homma and Saltelli[33], Sobol[34], Saltelli[35], Saltelli *et al.*[36], Saltelli *et al.*[37], Cukier *et al.*[38], Saltelli *et al.*[39], & Tarantola *et al.*[40] Saltelli *et al.*[41], Janon *et al.*[42], Owen[43], Tissot and Prieur[44], Da Veiga and Gamboa[45], Archer *et al.*[46], Tarantola *et al.*[47], Saltelli *et al.*[41] and Jansen[48]) and • exploring the model behaviour over a range on input values (Storlie and Helton[49] and Da Veiga *et al.*[50], Li *et al.*[51] and Hajikolaei and Wang[52]). Iooss and Lemaître[22] also provide various criteria in a flowchart for adapting a method or a combination of the methods for sensitivity analysis.

Besides the above Sobol'[21]'s variance based indicies, more recent developments regarding new indicies based on density, derivative and goal-oriented can be found in Borgonovo[53], Sobol and Kucherenko[54] and Fort *et al.*[55], respectively. In a more recent development, Da Veiga[56] propose new class of indicies based on density ratio estimation (Borgonovo[53]) that are special cases of dependence measures. This in turn helps in exploiting measures like distance correlation (Székely *et al.*[57]) and Hilbert-Schmidt independence criterion (Gretton *et al.*[58]) as new sensitivity indicies. The basic framework of these indicies is based on use of Csiszár *et al.*[59] f-divergence, concept of dissimilarity measure and kernel trick Aizerman *et al.*[60]. Finally, Da Veiga[56] propose feature selection as an alternative to screening methods in sensitivity analysis. The main issue with variance based indicies (Sobol'[21]) is that even though they capture importance information regarding the contribution of the input factors, they • do not handle multivariate random variables easily and • are only invariant under linear transformations. In comparison to these variance methods, the newly proposed indicies based on density estimations (Borgonovo[53]) and dependence measures are more robust.

2.2 Relevance in systems biology

Recent efforts in systems biology to understand the importance of various factors apropos output behaviour has gained prominence. Sumner *et al.*[61] compares the use of Sobol'[21] variance based indices versus Morris[23] screening method which uses a One-at-a-time (OAT) approach to analyse the sensitivity of $GSK3$ dynamics to uncertainty in an insulin signaling model. Similar efforts, but on different pathways can be found in Zheng and Rundell[62] and Marino *et al.*[63].

SA provides a way of analyzing various factors taking part in a biological phenomena and deals with the effects of these factors on the output of the biological system under consideration. Usually, the model equations are differential in nature with a set of inputs and the associated set of parameters that guide the output. SA helps in observing how the variance in these parameters and inputs leads to changes in the output behaviour. The goal of this manuscript is not to analyse differential equations and the parameters associated with it. Rather, the aim is to observe which input genotypic factors have greater contribution to observed phenotypic behaviour like a sample being normal or cancerous in both static and time series data. In this process, the effect of fold changes in time is also considered for analysis in the light of the recently observed psychophysical laws acting downstream of the Wnt pathway (Goentoro and Kirschner[64]).

2.3 Sensitivity indices

Given the range of estimators available for testing the sensitivity, it might be useful to list a few which are going to be employed in this research study. Also, a brief introduction into the fundamentals of the derivation of the three main indicies has been provided.

2.3.1 Variance based indices

The variance based indices as proposed by Sobol'[21] prove a theorem that an integrable function can be decomposed into summands of different dimensions. Also, a Monte Carlo algorithm is used to estimate the sensitivity of a function apropos arbitrary group of variables. It is assumed that a model denoted by function $u = f(x)$, $x = (x_1, x_2, \ldots, x_n)$, is defined in a unit n-dimensional cube $\mathcal{K}^n$ with u as the scalar output. The requirement of the problem is to find the sensitivity of function $f(x)$ with respect to different variables. If $u^* = f(x^*)$ is the required solution, then the sensitivity of u^* apropos x_k is estimated via the partial derivative $(\partial u/\partial x_k)_{x=x^*}$. This approach is the local sensitivity.

In global sensitivity, the input $x = x^*$ is not specified. This implies that the model $f(x)$ lies inside the cube and the sensitivity indices are regarded as tools for studying the model instead of the solution. Detailed technical aspects with examples can be found in Homma and Saltelli[33] and Sobol[65].

Let a group of indices $i_1, i_2, \ldots, i_s$ exist, where $1 \leq i_1 < \ldots < i_s \leq n$ and $1 \leq s \leq n$. Then the notation for sum over all different groups of indices is -

$$\widehat{\Sigma} T_{i_1,i_2,\ldots,i_s} = \Sigma_{i=1}^{n} T_i + \Sigma_{s=1}^{n} \Sigma_{1 \leq i < j \leq n} T_{i,j} + \ldots + T_{1,2,\ldots,n} \quad (1)$$

Then the representation of $f(x)$ using equation 1 in the form -

$$\begin{aligned} f(x) &= f_0 + \widehat{\Sigma} f_{i_1,i_2,\ldots,i_s} \\ &= f_0 + \Sigma_i f_i(x_i) + \Sigma_{i<j} f_{i,j}(x_i, x_j) + \ldots + f_{1,2,\ldots,n}(x_1, x_2, \ldots, x_n) \end{aligned} \quad (2)$$

is called ANOVA-decomposition from Archer *et al.*[46] or expansion into summands of different dimensions, if f_0 is a constant and integrals of the summands $f_{i_1,i_2,\ldots,i_s}$ with respect to their own variables are zero, i.e,

$$f_0 = \int_{\mathcal{K}^n} f(x) dx \quad (3)$$

$$\int_0^1 f_{i_1,i_2,\ldots,i_s}(x_{i_1}, x_{i_2}, \ldots, x_{i_s}) dx_{i_k} = 0, 1 \leq k \leq s \quad (4)$$

It follows from equation 3 that all summands on the right hand side are orthogonal, i.e if at least one of the indices in $i_1, i_2, \ldots, i_s$ and $j_1, j_2, \ldots, j_l$ is not repeated i.e

$$\int_0^1 f_{i_1,i_2,\ldots,i_s}(x_{i_1}, x_{i_2}, \ldots, x_{i_s}) f_{j_1,j_2,\ldots,j_l}(x_{j_1}, x_{j_2}, \ldots, x_{j_s}) dx = 0 \quad (5)$$

Sobol'[21] proves a theorem stating that there is an existence of a unique expansion of equation 3 for any $f(x)$ integrable in $\mathcal{K}^n$. In brief, this implies that for each of the indices as well as a group of indices, integrating equation 3 yields the following -

$$\int_0^1 \ldots \int_0^1 f(x) dx / dx_i = f_0 + f_i(x_i) \quad (6)$$

$$\int_0^1 \ldots \int_0^1 f(x) dx / dx_i dx_j = f_0 + f_i(x_i) + f_j(x_j) + f_{i,j}(x_i, x_j) \quad (7)$$

were, dx/dx_i is $\prod_{\forall k \in \{1,..,n\}; i \notin k} dx_k$ and $dx/dx_i dx_j$ is $\prod_{\forall k \in \{1,..,n\}; i,j \notin k} dx_k$. For higher orders of grouped indices, similar computations follow. The computation of any summand $f_{i_1,i_2,\ldots,i_s}(x_{i_1}, x_{i_2}, \ldots, x_{i_s})$ is reduced to an integral in the cube $\mathcal{K}^n$. The last summand $f_{1,2,\ldots,n}(x_1, x_2, \ldots, x_n)$ is $f(x) - f_0$ from equation 3. Homma and Saltelli[33] stresses that use of Sobol sensitivity indices does not require evaluation of any $f_{i_1,i_2,\ldots,i_s}(x_{i_1}, x_{i_2}, \ldots, x_{i_s})$ nor the knowledge of the form of $f(x)$ which might well be represented by a computational model i.e a function whose value is only obtained as the output of a computer program.

Finally, assuming that $f(x)$ is square integrable, i.e $f(x) \in \mathcal{L}_2$, then all of $f_{i_1,i_2,\ldots,i_s}(x_{i_1}, x_{i_2}, \ldots, x_{i_s}) \in \mathcal{L}_2$. Then the following constants

$$\int_{\mathcal{K}^n} f^2(x) dx - f_0^2 = D \quad (8)$$

$$\int_0^1 \ldots \int_0^1 f^2_{i_1,i_2,\ldots,i_s}(x_{i_1}, x_{i_2}, \ldots, x_{i_s}) dx_{i_1} dx_{i_2} \ldots dx_{i_s} = D_{i_1,i_2,\ldots,i_s} \quad (9)$$

are termed as variances. Squaring equation 3, integrating over $\mathcal{K}^n$ and using the orthogonality property in equation 5, D evaluates to -

$$D = \widehat{\Sigma} D_{i_1,i_2,\ldots,i_s} \quad (10)$$

Then the global sensitivity estimates is defined as -

$$S_{i_1,i_2,\ldots,i_s} = \frac{D_{i_1,i_2,\ldots,i_s}}{D} \quad (11)$$

It follows from equations 10 and 11 that

$$\widehat{\Sigma} S_{i_1,i_2,\ldots,i_s} = 1 \quad (12)$$

Clearly, all sensitivity indices are non-negative, i.e an index $S_{i_1,i_2,\ldots,i_s} = 0$ if and only if $f_{i_1,i_2,\ldots,i_s} \equiv 0$. The true potential of Sobol indices is observed when variables $x_1, x_2, \ldots, x_n$ are divided into m different groups with $y_1, y_2, \ldots, y_m$ such that $m < n$. Then $f(x) \equiv f(y_1, y_2, \ldots, y_m)$. All properties remain the same for the computation of sensitivity indices with the fact that integration with respect to y_k means integration with respect to all the x_i's in y_k. Details of these computations with examples can be found in Sobol[65]. Variations and improvements over Sobol indices have already been stated in section 2.1.

2.3.2 Density based indices

As discussed before, the issue with variance based methods is the high computational cost incurred due to the number of interactions among the variables. This further requires the use of screening methods to filter out redundant or unwanted factors that might not have significant impact on the output. Recent work by Da Veiga[56] proposes a new class of sensitivity indicies which are a special case of density based indicies Borgonovo[53]. These indicies can handle multivariate variables easily and relies on density ratio estimation. Key points from Da Veiga[56] are mentioned below.

Considering the similar notation in previous section, $f : \mathcal{R}^n \rightarrow \mathcal{R}$ ($u = f(x)$) is assumed to be continuous. It is also assumed that X_k has a known distribution and are independent. Baucells and

Borgonovo[66] state that a function which measures the similarity between the distribution of U and that of $U|X_k$ can define the impact of X_k on U. Thus the impact is defined as -

$$S_{X_k} = \mathcal{E}(d(U, U|X_k)) \tag{13}$$

were $d(\cdot,\cdot)$ is a dissimilarity measure between two random variables. Here d can take various forms as long as it satisfies the criteria of a dissimilarity measure. Csiszár *et al.*[59]'s f-divergence between U and $U|X_k$ when all input random variables are considered to be absolutely continuous with respect to Lebesgue measure on $\mathcal{R}$ is formulated as -

$$d_F(U||U|X_k) = \int_{\mathcal{R}} F(\frac{p_U(u)}{p_{U|X_k}(u)}) p_{U|X_k}(u) du \tag{14}$$

were F is a convex function such that $F(1) = 0$ and p_U and $p_{U|X_k}$ are the probability distribution functions of U and $U|X_k$. Standard choices of F include Kullback-Leibler divergence $F(t) = -\log_e(t)$, Hellinger distance $(\sqrt{t}-1)^2$, Total variation distance $F(t) = |t-1|$, Pearson χ^2 divergence $F(t) = t^2 - 1$ and Neyman χ^2 divergence $F(t) = (1-t^2)/t$. Substituting equation 14 in equation 13, gives the following sensitivity index -

$$\begin{aligned} S^F_{X_k} &= \int_{\mathcal{R}} d_F(U||U|X_k) p_{X_k}(x) dx \\ &= \int_{\mathcal{R}} \int_{\mathcal{R}} F(\frac{p_U(u)}{p_{U|X_k}(u)}) p_{U|X_k}(u) p_{X_k}(x) dx du \\ &= \int_{\mathcal{R}^2} F(\frac{p_U(u) p_{X_k}(x)}{p_{U|X_k}(u) p_{X_k}(x)}) p_{U|X_k}(u) p_{X_k}(x) dx du \\ &= \int_{\mathcal{R}^2} F(\frac{p_U(u) p_{X_k}(x)}{p_{X_k,U}(x,u)}) p_{X_k,U}(x,u) dx du \end{aligned} \tag{15}$$

were p_{X_k} and $p_{X_k,Y}$ are the probability distribution functions of X_k and (X_k, U), respectively. Csiszár *et al.*[59] f-divergences imply that these indices are positive and equate to 0 when U and X_k are independent. Also, given the formulation of $S^F_{X_k}$, it is invariant under any smooth and uniquely invertible transformation of the variables X_k and U (Kraskov *et al.*[67]). This has an advantage over Sobol sensitivity indices which are invariant under linear transformations.

By substituting the different formulations of F in equation 15, Da Veiga[56]'s work claims to be the first in establishing the link that previously proposed sensitivity indices are actually special cases of more general indices defined through Csiszár *et al.*[59]'s f-divergence. Then equation 15 changes to estimation of ratio between the joint density of (X_k, U) and the marginals, i.e -

$$S^F_{X_k} = \int_{\mathcal{R}^2} F(\frac{1}{r(x,u)}) p_{X_k,U}(x,u) dx du = \mathcal{E}_{(X_k,U)} F(\frac{1}{r(X_k,U)}) \tag{16}$$

were, $r(x,y) = (p_{X_k,U}(x,u))/(p_U(u) p_{X_k}(x))$. Multivariate extensions of the same are also possible under the same formulation.

Finally, given two random vectors $X \in \mathcal{R}^p$ and $Y \in \mathcal{R}^q$, the dependence measure quantifies the dependence between X and Y with the property that the measure equates to 0 if and only if X and Y are independent. These measures carry deep links (Sejdinovic *et al.*[68]) with distances between embeddings of distributions to reproducing kernel Hilbert spaces (RHKS) and here the related Hilbert-Schmidt independence criterion (HSIC by Gretton *et al.*[58]) is explained.

In a very brief manner from an extremely simple introduction by Daumé III[69] - "We first defined a field, which is a space that supports the usual operations of addition, subtraction, multiplication and division. We imposed an ordering on the field and described what it means for a field to be complete. We then defined vector spaces over fields, which are spaces that interact in a friendly way with their associated fields. We defined complete vector spaces and extended them to Banach spaces by adding a norm. Banach spaces were then extended to Hilbert spaces with the addition of a dot product." Mathematically, a Hilbert space $\mathcal{H}$ with elements $r, s \in \mathcal{H}$ has dot product $\langle r, s\rangle_{\mathcal{H}}$ and $r \cdot s$. When $\mathcal{H}$ is a vector space over a field $\mathcal{F}$, then the dot product is an element in $\mathcal{F}$. The product $\langle r, s\rangle_{\mathcal{H}}$ follows the below mentioned properties when $r, s, t \in \mathcal{H}$ and for all $a \in \mathcal{F}$ -

- Associative : $(ar) \cdot s = a(r \cdot s)$
- Commutative : $r \cdot s = s \cdot r$
- Distributive : $r \cdot (s+t) = r \cdot s + r \cdot t$

Given a complete vector space $\mathcal{V}$ with a dot product $\langle \cdot, \cdot \rangle$, the norm on $\mathcal{V}$ defined by $||r||_{\mathcal{V}} = \sqrt{(\langle r, r\rangle)}$ makes this space into a Banach space and therefore into a full Hilbert space.

A reproducing kernel Hilbert space (RKHS) builds on a Hilbert space $\mathcal{H}$ and requires all Dirac evaluation functionals in $\mathcal{H}$ are bounded and continuous (on implies the other). Assuming $\mathcal{H}$ is the $\mathcal{L}_2$ space of functions from X to $\mathcal{R}$ for some measurable X. For an element $x \in X$, a Dirac evaluation functional at x is a functional $\delta_x \in \mathcal{H}$ such that $\delta_x(g) = g(x)$. For the case of real numbers, x is a vector and g a function which maps from this vector space to $\mathcal{R}$. Then δ_x is simply a function which maps g to the value g has at x. Thus, δ_x is a function from $(\mathcal{R}^n \mapsto \mathcal{R})$ into $\mathcal{R}$.

The requirement of Dirac evaluation functions basically means (via the Riesz[70] representation theorem) if ϕ is a bounded linear

functional (conditions satisfied by the Dirac evaluation functionals) on a Hilbert space $\mathcal{H}$, then there is a unique vector ℓ in $\mathcal{H}$ such that $\phi g = \langle g, \ell \rangle_{\mathcal{H}}$ for all $\ell \in \mathcal{H}$. Translating this theorem back into Dirac evaluation functionals, for each δ_x there is a unique vector k_x in $\mathcal{H}$ such that $\delta_x g = g(x) = \langle g, k_x \rangle_{\mathcal{H}}$. The reproducing kernel K for $\mathcal{H}$ is then defined as : $K(x, x') = \langle k_x, k_{x'} \rangle$, were k_x and $k_{x'}$ are unique representatives of δ_x and $\delta_{x'}$. The main property of interest is $\langle g, K(x, x') \rangle_{\mathcal{H}} = g(x')$. Furthermore, k_x is defined to be a function $y \mapsto K(x, y)$ and thus the reproducibility is given by $\langle K(x, \cdot), K(y, \cdot) \rangle_{\mathcal{H}} = K(x, y)$.

Basically, the distance measures between two vectors represent the degree of closeness among them. This degree of closeness is computed on the basis of the discriminative patterns inherent in the vectors. Since these patterns are used implicitly in the distance metric, a question that arises is, how to use these distance metric for decoding purposes?

The kernel formulation as proposed by Aizerman *et al.*[60], is a solution to our problem mentioned above. For simplicity, we consider the labels of examples as binary in nature. Let $\mathbf{x}_i \in \mathcal{R}^n$, be the set of n feature values with corresponding category of the example label (y_i) in data set $\mathcal{D}$. Then the data points can be mapped to a higher dimensional space $\mathcal{H}$ by the transformation ϕ:

$$\phi : \mathbf{x}_i \in \mathcal{R}^n \mapsto \phi(\mathbf{x}_i) \in \mathcal{H} \tag{17}$$

This $\mathcal{H}$ is the *Hilbert Space* which is a strict inner product space, along with the property of completeness as well as separability. The inner product formulation of a space helps in discriminating the location of a data point w.r.t a separating hyperplane in $\mathcal{H}$. This is achieved by the evaluation of the inner product between the normal vector representing the hyperplane along with the vectorial representation of a data point in $\mathcal{H}$. Thus, the idea behind equation(17) is that even if the data points are nonlinearly clustered in space $\mathcal{R}^n$, the transformation spreads the data points into $\mathcal{H}$, such that they can be linearly separated in its range in $\mathcal{H}$.

Often, the evaluation of dot product in higher dimensional spaces is computationally expensive. To avoid incurring this cost, the concept of kernels in employed. The trick is to formulate kernel functions that depend on a pair of data points in the space $\mathcal{R}^n$, under the assumption that its evaluation is equivalent to a dot product in the higher dimensional space. This is given as:

$$\kappa(\mathbf{x}_i, \mathbf{x}_j) = < \phi(\mathbf{x}_i), \phi(\mathbf{x}_j) > \tag{18}$$

Two advantages become immediately apparent. First, the evaluation of such kernel functions in lower dimensional space is computationally less expensive than evaluating the dot product in higher dimensional space. Secondly, it relieves the burden of searching an appropriate transformation that may map the data points in $\mathcal{R}^n$ to $\mathcal{H}$. Instead, all computations regarding discrimination of location of data points in higher dimensional space involves evaluation of the kernel functions in lower dimension. The matrix containing these kernel evaluations is referred to as the *kernel* matrix. With a cell in the kernel matrix containing a kernel evaluation between a pair of data points, the kernel matrix is square in nature.

As an example in practical applications, once the kernel has been computed, a pattern analysis algorithm uses the kernel function to evaluate and predict the nature of the new example using the general formula:

$$\begin{aligned} f(\mathbf{z}) &= < \mathbf{w}, \phi(\mathbf{z}) > + b \\ &= < \sum_{i=1}^{N} \alpha_i \times y_i \times \phi(\mathbf{x}_i), \phi(\mathbf{z}) > + b \\ &= \sum_{i=1}^{N} \alpha_i \times y_i \times < \phi(\mathbf{x}_i), \phi(\mathbf{z}) > + b \\ &= \sum_{i=1}^{N} \alpha_i \times y_i \times \kappa(\mathbf{x}_i, \mathbf{z}) + b \end{aligned} \tag{19}$$

where $\mathbf{w}$ defines the hyperplane as some linear combination of training basis vectors, $\mathbf{z}$ is the test data point, y_i the class label for training point $\mathbf{x}_i$, α_i and b are the constants. Various transformations to the kernel function can be employed, based on the properties a kernel must satisfy. Interested readers are referred to Taylor and Cristianini[71] for description of these properties in detail.

The Hilbert-Schmidt independence criterion (HSIC) proposed by Gretton *et al.*[58] is based on kernel approach for finding dependences and on cross-covariance operators in RKHS. Let $X \in \mathcal{X}$ have a distribution P_X and consider a RKHS $\mathcal{A}$ of functions $\mathcal{X} \to \mathcal{R}$ with kernel $k_{\mathcal{X}}$ and dot product $\langle \cdot, \cdot \rangle_{\mathcal{A}}$. Similarly, Let $U \in \mathcal{Y}$ have a distribution P_Y and consider a RKHS $\mathcal{B}$ of functions $\mathcal{U} \to \mathcal{R}$ with kernel $k_{\mathcal{B}}$ and dot product $\langle \cdot, \cdot \rangle_{\mathcal{B}}$. Then the cross-covariance operator $C_{X,U}$ associated with the joint distribution P_{XU} of (X, U) is the linear operator $\mathcal{B} \to \mathcal{A}$ defined for every $a \in \mathcal{A}$ and $b \in \mathcal{B}$ as -

$$\langle a, C_{XU} b \rangle_{\mathcal{A}} = \mathcal{E}_{XU}[a(X), b(U)] - \mathcal{E}_X a(X) \mathcal{E}_U b(U) \tag{20}$$

The cross-covariance operator generalizes the covariance matrix by representing higher order correlations between X and U through nonlinear kernels. For every linear operator $C : \mathcal{B} \to \mathcal{A}$ and provided the sum converges, the Hilbert-Schmidt norm of C is given by -

$$||C||_{HS}^2 = \Sigma_{k,l} \langle a_k, C b_l \rangle_{\mathcal{A}} \tag{21}$$

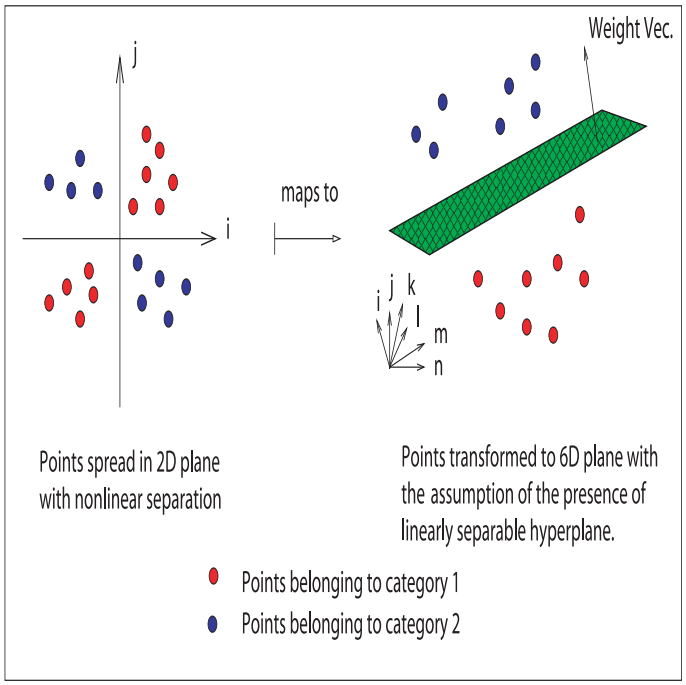

Fig. 2 A geometrical interpretation of mapping nonlinearly separable data into higher dimensional space where it is assumed to be linearly separable, subject to the holding of dot product.

were a_k and b_l are orthonormal bases of $\mathcal{A}$ and $\mathcal{B}$, respectively. The HSIC criterion is then defined as the Hilbert-Schmidt norm of cross-covariance operator -

$$HSIC(X,U)_{\mathcal{A},\mathcal{B}} = \begin{cases} ||C_{XU}||^2_{HS} = \\ \mathcal{E}_{X,X',U,U'} k_{\mathcal{X}}(X,X') k_{\mathcal{U}}(U,U') + \\ \mathcal{E}_{X,X'} k_{\mathcal{X}}(X,X') \mathcal{E}_{U,U'} k_{\mathcal{U}}(U,U') - \\ 2\mathcal{E}_{X,Y}[\mathcal{E}_{X'} k_{\mathcal{X}}(X,X') \mathcal{E}_{U'} k_{\mathcal{U}}(U,U')] \end{cases} \tag{22}$$

were the equality in terms of kernels is proved in Gretton *et al.*[58]. Finally, assuming (X_i, U_i) $(i = 1, 2, \ldots, n)$ is a sample of the random vector (X, U) and denote $K_{\mathcal{X}}$ and $K_{\mathcal{U}}$ the Gram matrices with entries $K_{\mathcal{X}}(i,j) = k_{\mathcal{X}}(X_i, X_j)$ and $K_{\mathcal{U}}(i,j) = k_{\mathcal{U}}(U_i, U_j)$. Gretton *et al.*[58] proposes the following estimator for $HSIC_n(X,U)_{\mathcal{A},\mathcal{B}}$ -

$$HSIC_n(X,U)_{\mathcal{A},\mathcal{B}} = \frac{1}{n^2} Tr(K_{\mathcal{X}} H K_{\mathcal{U}} H) \tag{23}$$

were H is the centering matrix such that $H(i,j) = \delta_{i,j} - \frac{1}{n}$. Then $HSIC_n(X,U)_{\mathcal{A},\mathcal{B}}$ can be expressed as -

$$HSIC(X,U)_{\mathcal{A},\mathcal{B}} = \begin{cases} \frac{1}{n^2}\Sigma_{i,j=1}^n k_{\mathcal{X}}(X_i,X_j) k_{\mathcal{U}}(U_i,U_j) \\ + \frac{1}{n^2}\Sigma_{i,j=1}^n k_{\mathcal{X}}(X_i,X_j) \frac{1}{n^2}\Sigma_{i,j=1}^n k_{\mathcal{U}}(U_i,U_j) \\ - \frac{2}{n}\Sigma_{i=1}^n [\frac{1}{n}\Sigma_{j=1}^n k_{\mathcal{X}}(X_i,X_j) \frac{1}{n}\Sigma_{j=1}^n k_{\mathcal{U}}(U_i,U_j)] \end{cases} \tag{24}$$

Finally, Da Veiga[56] proposes the sensitivity index based on distance correlation as -

$$S_{X_k}^{HSIC_{\mathcal{A},\mathcal{B}}} = R(X_k, U)_{\mathcal{A},\mathcal{B}} \tag{25}$$

were the kernel based distance correlation is given by -

$$R^2(X,U)_{\mathcal{A},\mathcal{B}} = \frac{HSIC(X,U)_{\mathcal{A},\mathcal{B}}}{\sqrt{(HSIC(X,X)_{\mathcal{A},\mathcal{A}} HSIC(U,U)_{\mathcal{B},\mathcal{B}})}} \tag{26}$$

were kernels inducing $\mathcal{A}$ and $\mathcal{B}$ are to be chosen within a universal class of kernels. Similar multivariate formulation for equation 23 are possible.

2.3.3 Choice of sensitivity indices

The SENSITIVITY PACKAGE (Faivre *et al.*[72] and Iooss and Lemaître[22]) in R langauge provides a range of functions to compute the indices and the following indices will be taken into account for addressing the posed questions in this manuscript.

1. sensiFdiv - conducts a density-based sensitivity analysis where the impact of an input variable is defined in terms of dissimilarity between the original output density function and the output density function when the input variable is fixed. The dissimilarity between density functions is measured with Csiszar f-divergences. Estimation is performed through kernel density estimation and the function kde of the package ks. (Borgonovo[53], Da Veiga[56])

2. sensiHSIC - conducts a sensitivity analysis where the impact of an input variable is defined in terms of the distance between the input/output joint probability distribution and the product of their marginals when they are embedded in a Reproducing Kernel Hilbert Space (RKHS). This distance corresponds to HSIC proposed by Gretton *et al.*[58] and serves as a dependence measure between random variables.

3. soboljansen - implements the Monte Carlo estimation of the Sobol indices for both first-order and total indices at the same time (all together 2p indices), at a total cost of (p+2) × n model evaluations. These are called the Jansen estimators. (Jansen[48] and Saltelli *et al.*[41])

4. sobol2002 - implements the Monte Carlo estimation of the Sobol indices for both first-order and total indices at the same time (all together 2p indices), at a total cost of (p+2) ×n model evaluations. These are called the Saltelli estimators. This estimator suffers from a conditioning problem when estimating the variances behind the indices computations. This can seriously affect the Sobol indices estimates in case of largely non-centered output. To avoid this effect, you have to center the model output before applying

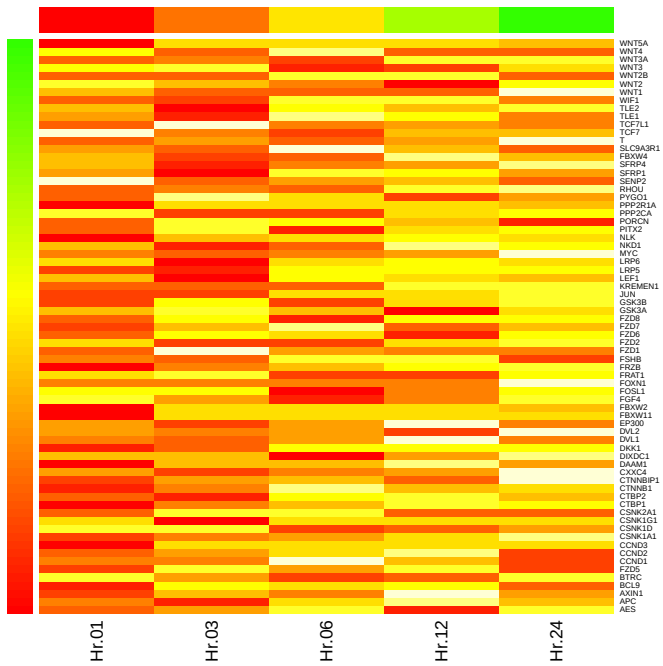

Fig. 3 Heat map for gene expression values for 6 time points from Gujral and MacBeath[17]

"sobol2002". Functions "soboljansen" and "sobolmartinez" do not suffer from this problem. (Saltelli[35])

5. sobol2007 - implements the Monte Carlo estimation of the Sobol indices for both first-order and total indices at the same time (all together 2p indices), at a total cost of (p+2) × n model evaluations. These are called the Mauntz estimators. (Saltelli *et al.*[41])

6. sobolmartinez - implements the Monte Carlo estimation of the Sobol indices for both first-order and total indices using correlation coefficients-based formulas, at a total cost of (p + 2) × n model evaluations. These are called the Martinez estimators.

7. sobol - implements the Monte Carlo estimation of the Sobol sensitivity indices. Allows the estimation of the indices of the variance decomposition up to a given order, at a total cost of (N + 1) × n where N is the number of indices to estimate. (Sobol'[21])

3 Optimization and Support Vector Machines

Aspects of SVMs from Sinha[73] are reproduced for completeness.

3.1 Optimization Problems

3.1.1 Introduction

The main focus in this section is optimization problems, the concept of Lagrange multipliers and KKT conditions, which will be later used to explain the details about the SVMs.

3.1.2 Mathematical Formulation

Optimization problems arise in almost every area of engineering. The goal is to achieve an almost perfect and efficient result, while carrying out certain procedures of optimization. Our main source of reference on this topic derives from Bletzinger[74]. We will be using notations used in Bletzinger[74]. In mathematical terms the general form of optimization problem can be represented as :

$$\begin{aligned} \text{minimize} \quad &: \quad f(\mathbf{x});\ \mathbf{x} \in \mathbf{R}^n \\ \text{such that} \quad &: \quad g_j(\mathbf{x}) \leq 0;\ \mathrm{j} = 1,\ldots\ldots,\mathrm{p} \\ &: \quad h_j(\mathbf{x}) = 0;\ \mathrm{j} = 1,\ldots\ldots,\mathrm{q}. \end{aligned} \tag{27}$$

where f, g_j and h_j are the objective function, equality constraints and inequality constraints. Generally, the number of constraints is less than the number of variables used to formulate the optimization problem. For a problem to be linear, both the constraints and the objective function need to be linear. Quadratic problems require only the objective function to be quadratic, while the constraints remain linear in formulation. Besides these, if any one of the functions is nonlinear, then the problem becomes nonlinear in nature. A graphical view of the types of the problems can be seen in fig. 4).

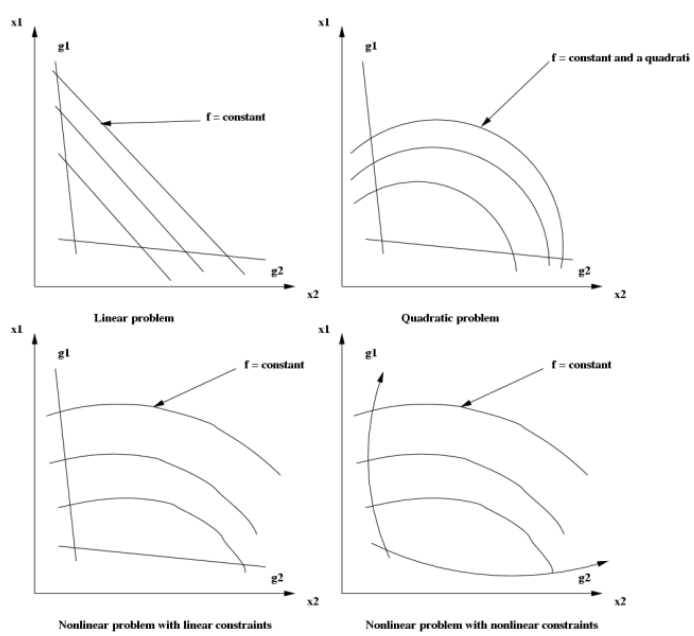

Fig. 4 Kinds of optimization problems.

3.1.3 Lagrange Multipliers

In unconstrained optimization problems, where the first order derivatives are assumed continuous, the solution is found by solving:

$$\nabla_x f = \frac{\partial f}{\partial x_i} = 0; \text{ i = 1,.....,n.} \tag{28}$$

where f is a function of x. Since most of the optimization problems are constrained, the concept of Lagrange multipliers is introduced in order to solve the problem. Thus, the Lagrangian formulation, for Eqn. 27 becomes:

$$L(\mathbf{x}, \lambda, \mu) = f(\mathbf{x}) + \sum_{j=1}^{p} \lambda_j g_j(\mathbf{x}) + \sum_{j=1}^{q} \mu_j h_j(\mathbf{x}) \tag{29}$$

where L is the Lagrangian, λ and μ are the vectors of the Lagrange multipliers for inequality and equality constraints, respectively.

Next comes the solving of the Lagrangian. We try to derive a solution in terms of variables used and show that the final solution achieved by Equ. 27 and Eqn. 29 remains the same. For the sake of derivation, we assume that each of the vectors x, λ and μ have a single element and also there exists a single optimal solution. We will then generalize the solution to vectors containing various elements. Let x^*, λ^* and μ^* be the optimal solution for the Lagrangian. Let $x^!$ be the optimal solution for $f(x)$. To begin with, our Lagrangian has the form:

$$L(x, \lambda, \mu) = f(x) + \lambda g(x) + \mu h(x) \tag{30}$$

Derivation:

- **Step 1:** Differentiate the Lagrangian in Eqn 30 w.r.t x and equate it to zero.

$$\frac{\partial L}{\partial x} = \frac{\partial f}{\partial x} + \lambda \frac{\partial g}{\partial x} + \mu \frac{\partial h}{\partial x} = 0 \tag{31}$$

- **Step 2:** Find x in terms of λ and μ, such that $x = x(\lambda, \mu)$.

- **Step 3:** Differentiate the Lagrangian in Eqn. 30 w.r.t λ and equate it to zero.

$$\frac{\partial L}{\partial \lambda} = g(x) = 0 \tag{32}$$

- **Step 4:** Differentiate the Lagrangian in Eqn 30 w.r.t μ and equate it to zero.

$$\frac{\partial L}{\partial \mu} = h(x) = 0 \tag{33}$$

- **Step 5:** Substitute $x(\lambda, \mu)$ in Equ. 32 and Eqn. 33 to get two equations in two unknowns λ and μ and solve to get the optimal values.

$$g(x(\lambda, \mu)) = 0 \tag{34}$$

$$h(x(\lambda, \mu)) = 0 \tag{35}$$

Let λ^*, μ^* be the solution. Substituting these in $x = x(\lambda, \mu)$, we get x^*.

- **Step 6:** Combining Eqn. 32 and Eqn. 33 in Eqn. 30, along with λ^*, μ^* and x^*, we have:

$$L(x^*, \lambda^*, \mu^*) = f(x^*) + \lambda^* g(x^*) + \mu^* h(x^*) = f(x^*) \tag{36}$$

Since, it is assumed that there exist only one optimal solution we have:

$$\begin{aligned} L(x^*, \lambda^*, \mu^*) &= f(x^*) = f(x^!) \\ x^* &= x^! \end{aligned} \tag{37}$$

Lastly, since $g(x)$ in Eqn. 32 is a inequality constraint, we have:

$$\begin{aligned} \lambda g(x) &= 0 \\ \lambda &\geq 0 \end{aligned} \tag{38}$$

3.1.4 Dual Functions

For sake of simplicity, let us for a moment ignore the equality constraint. Then the Lagrangian becomes:

$$L(x, \lambda, \mu) = f(x) + \lambda g(x). \tag{39}$$

It is sometimes easy to transform the Lagrangian into a simpler form, in order to find an optimal solution. We can represent the Lagrangian as a Dual function in such a manner that the optimal solution defined as minimum of $L(x, \lambda^*)$ w.r.t x where $\lambda = \lambda^*$, can be represented as the maximum of dual function $D(\lambda)$ w.r.t λ. For a given λ, the dual is evaluated by finding the minimum of $L(x, \lambda)$ w.r.t x. Thus to find the optimal point we evaluate:

$$\max_{\lambda} D(\lambda) = \min_{x} L(x, \lambda^*) \tag{40}$$

So the basic steps to solve the dual problem are as follows: **Step 1:** Minimize $L(x, \lambda)$ w.r.t x, and find x in terms of λ. **Step 2:** Substitute $x(\lambda)$ in L s.t. $D(\lambda) = L(x(\lambda), \lambda)$. **Step 3:** Maximize $D(\lambda)$ w.r.t λ.

3.1.5 Karush Kuhn Tucker Conditions

The derivation in the last part (Eqn. 30 to Eqn. 38)gives us a set of equations that need to be evaluated along with the consideration of constraints present. These set of equations and constraints in terms of the Lagrangian, form the Karush Kuhn Tucker Conditions. We give here the generalized KKT conditions and explain

the necessary details.

$$\begin{array}{lll} \frac{\partial L}{\partial x_i} = \frac{\partial f}{\partial x_i} + \sum_{j=1}^{p} \lambda_j \frac{\partial g_j}{\partial x_i} + \sum_{j=1}^{p} \mu_j \frac{\partial h_j}{\partial x_i} = 0 & : & i = 1, \ldots, n \\ \frac{\partial L}{\partial \lambda_j} = g_j(\mathbf{x}) = 0 & : & j = 1, \ldots, p \\ \frac{\partial L}{\partial \mu_j} = h_j(\mathbf{x}) = 0 & : & j = 1, \ldots, q \\ \lambda_j g_j = 0 & : & j = 1, \ldots, p \\ \lambda_j \geq 0 & : & j = 1, \ldots, p \end{array} \tag{41}$$

where L is Eqn. 29.

The KKT conditions specify a few points which are as follows:

1. The first line states that the linear combination of objective and constraint gradients vanishes.

2. A prerequisite of the KKT conditions is that the gradients of the constraints must be continuous (evident from second and third lines in Eqn. 41).

3. The last two lines in Eqn. 41 state that at optimum either the constraints are active or the constrains are inactive.

3.2 Support Vector Machines

Armed with the knowledge of optimization problems and concept of Lagrange multipliers, we now delve into the workings of support vector machines. Burges[75] provides a good introduction to SVMs and is our main reference. Interested readers should refer to Cristianini and Shawe-Taylor[76], Schölkopf and Smola[77] and Vapnik and Vapnik[78] for detailed references.

3.2.1 Separable Case

Let us suppose that we are presented with a data set that is linearly separable. We assume that there are m examples of data in the format $\{\mathbf{x}_i, y_i\}$, s.t. $\mathbf{x}_i \in \mathbf{R}^n$; $i = 1, \ldots, m$, where $y_i \in \{-1, 1\}$ is the corresponding true label of $\mathbf{x}_i$. We also suppose there is an existence of a linear hyperplane in the n dimensional space that separates the positively labeled data from the negatively labeled data. Let this separating hyperplane be given by

$$\mathbf{w} \cdot \mathbf{x} + b = 0. \tag{42}$$

where, $\mathbf{w}$ is the normal vector $\perp$ to the hyperplane and $|b|/||\mathbf{w}||$ is the shortest perpendicular distance of the hyperplane to the origin. $||\mathbf{w}||$ is the Euclidean norm of $\mathbf{w}$. The *margin* of a hyperplane is then defined as the minimum of the distance of the positively and negatively labeled examples, to the hyperplane. For the linear case, the SVM searches for the hyperplane with largest margin. We now have three conditions, based on the location of

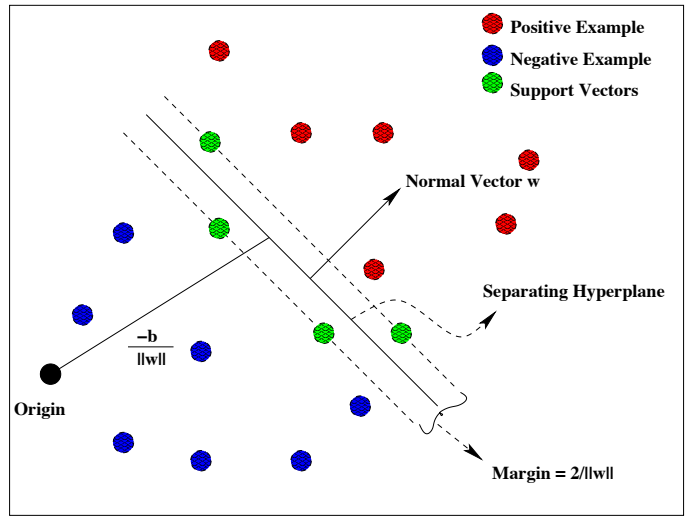

Fig. 5 Linear hyperplane for separable data. Adapted from Burges (A Tutorial on Support Vector Machines)

an example $\mathbf{x}_i$ w.r.t the hyperplane:

$$\mathbf{w} \cdot \mathbf{x}_i + b = 0 \quad : \quad \text{example lying on the hyperplane.} \tag{43}$$

$$\mathbf{w} \cdot \mathbf{x}_i + b \geq +1 \quad : \quad \text{positively labeled example.} \tag{44}$$

$$\mathbf{w} \cdot \mathbf{x}_i + b \leq -1 \quad : \quad \text{negatively labeled example.} \tag{45}$$

Combining the equality and the two inequalities we have:

$$y_i(\mathbf{w} \cdot \mathbf{x}_i + b) - 1 \geq 0 \tag{46}$$

Since the SVMs search for the largest margin, we now try to find a mathematical expression of the margin. Considering the examples that satisfy equality in Eqn. 44, the distance of the closest positive example can be expressed as $|1 - b|/||\mathbf{w}||$. Similarly, considering the negative examples that satisfy equality in Eqn. 45, the distance of the closest negative example can be expressed as $|-1 - b|/||\mathbf{w}||$. On summation of the two shortest distances, we get the margin of the hyperplane as $2/||\mathbf{w}||$. Since the labels are $\{-1, 1\}$, no example lies inside the hyperplanes representing the margin in this case. Taking into account that the SVM searches for the largest margin, we can say that it can be achieved by minimizing $||\mathbf{w}||^2$, subject to the constraints in Eqn. 46. Examples lying on the hyperplanes of the margins are termed *support vectors*, as their removal would change the margin and thus the solution. Figure 5 represents the conceptual points about separating hyperplanes.

3.3 Lagrangian Representation: Separable Case

Clearly, the previous paragraph shows that finding the margin is a problem of optimization as the goal is to minimize $||\mathbf{w}||^2$ subject to constraints in Eqn. 46. Employing the ideas of Chapter 3, the Lagrangian for the above problem, is:

$$L(\mathbf{w}, b, \alpha) = \frac{1}{2}||\mathbf{w}||^2 - \sum_{i=1}^{m} \alpha_i y_i (\mathbf{x}_i \cdot \mathbf{w} + b) + \sum_{i=1}^{m} \alpha_i \tag{47}$$

where $\frac{1}{2}||\mathbf{w}||^2$ is the objective function, α is the Lagrangian multiplier and the Eqn. 46 is the inequality constraint. Since the minimization of the objective function is required, we employ the ideas of the derivation of KKT conditions (Eqn. 41) to Eqn. 47. In short, we would require the $L(\mathbf{w}, b, \alpha)$ to be minimized w.r.t $\mathbf{w}$ and b and also require its derivative w.r.t all α_i's to vanish. Thus the KKT conditions take the form:

$$\begin{array}{lll}
\frac{\partial L}{\partial w_j} = w_j - \sum_i \alpha_i y_i x_{ij} = 0 & : & \text{j = 1,....,n} \\
\frac{\partial L}{\partial b} = -\sum_{i=1}^{m} \alpha_i y_i = 0 & : & \text{i = 1,....,m} \\
\frac{\partial L}{\partial \alpha_i} = -\sum_{i=1}^{m} y_i (\mathbf{x}_i \cdot \mathbf{w} + b) + \sum_{i=1}^{m} 1 = 0 & : & \text{i = 1,....,m} \\
\alpha_i (y_i (\mathbf{x}_i \cdot \mathbf{w} + b) - 1) = 0 & : & \text{i = 1,....,m} \\
\alpha_i \geq 0 & : & \text{i = 1,....,m}
\end{array} \tag{48}$$

Thus solving the SVMs is equivalent to solving the KKT conditions. While $\mathbf{w}$ is determined by the training set, b can be found by solving the penultimate equation in Eqn. 48 for which $\alpha_i \neq 0$. Also note that examples that have $\alpha_i \neq 0$ form the set of support vectors.

The dual problem for the same Lagrangian is:

$$D = \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j \tag{49}$$

Solving for Eqn. 49 requires maximization of D w.r.t α_i, subject to second line of Eqn. 48 and positivity of α_i, with the solution given by first line of Eqn. 48.

To classify or predict the label of a new example $\mathbf{x}_{new}$, the SVM has to evaluate $(\mathbf{x}_{new} \cdot \mathbf{w} + b)$ and check the sign of the evaluated value. A positive sign would lead to assignment of a $+1$ label and a negative sign to -1.

3.4 Nonseparable Case

For many classification problems, the data present is nonseparable. To extend the idea to nonseparable case, some amount of cost is added, which takes care of particular cases of examples. This is achieved by introducing slack in the constraints Eqn. 44 and Eqn. 45 (Burges[75], Vapnik and Vapnik[78]). The equations then becomes

$$\mathbf{w} \cdot \mathbf{x}_i + b \geq 1 - \xi_i \quad : \quad \text{positively labeled example.} \tag{50}$$

$$\mathbf{w} \cdot \mathbf{x}_i + b \leq -1 + \xi_i \quad : \quad \text{negatively labeled example.} \tag{51}$$

For an error to occur, the ξ_i value must exceed unity. To take care of the cost of errors, a penalty is introduced which changes the objective function from $||\mathbf{w}||^2/2$ to $||\mathbf{w}||^2/2 + C(\sum_i \xi_i)^k$. Thus $\sum_i \xi_i$ represents the upper bound on the training error. For quadratic problems, k can be 1 or 2.

3.5 Lagrangian Representation: Nonseparable Case

Since the formulation of the Lagrangian and its dual follow the same procedure, as mentioned before, we only mention the equations. The Lagrangian for nonlinear nonseparable case is:

$$\begin{aligned}
L(\mathbf{w}, b, \alpha, \xi) \quad = \quad & \frac{1}{2}||\mathbf{w}||^2 + C \sum_{i=1}^{m} \xi_i - \sum_{i=1}^{m} \alpha_i \{ y_i (\mathbf{x}_i \cdot \mathbf{w} + b) \\
& -1 + \xi_i \} - \sum_{i=1}^{m} \mu_i \xi_i
\end{aligned} \tag{52}$$

The corresponding KKT conditions are:

$$\begin{array}{lll}
\frac{\partial L}{\partial w_j} = w_j - \sum_i \alpha_i y_i x_{ij} = 0 & : & j = 1, \ldots, n \\
\frac{\partial L}{\partial b} = -\sum_{i=1}^{m} \alpha_i y_i = 0 & : & i = 1, \ldots, m \\
\frac{\partial L}{\partial \alpha_i} = y_i (\mathbf{x}_i \cdot \mathbf{w} + b) - 1 + \xi_i = 0 & : & i = 1, \ldots, m \\
\frac{\partial L}{\partial \xi_i} = C - \alpha_i - \mu_i = 0 & : & i = 1, \ldots, m \\
\alpha_i \{ y_i (\mathbf{x}_i \cdot \mathbf{w} + b) - 1 + \xi_i \} = 0 & : & i = 1, \ldots, m \\
\mu_i \xi_i = 0 & : & i = 1, \ldots, m \\
\alpha_i \geq 0 & : & i = 1, \ldots, m \\
\xi_i \geq 0 & : & i = 1, \ldots, m \\
\mu_i \geq 0 & : & i = 1, \ldots, m
\end{array} \tag{53}$$

The dual formulation $k = 1$ for the Lagrangian just discussed is:

$$D = \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j \tag{54}$$

All the previous conditions remain same, except that the Lagrangian multiplier α_i now has a upper bound of value C. The solution for the dual is given by $\mathbf{w} = \sum_{i=1}^{N_s} \alpha_i y_i \mathbf{x}_i$. N_s is the number of support vectors. Figure 6 depicts the nonseparable case.

3.6 Kernels and Space Dimensionality Transformation

The above cases were for linear separating hyperplanes. In order to generalize for nonlinear cases, Boser et.al[79] employed the idea of Aizerman[60] as follows; Since the Dual in Eqn. 54 and

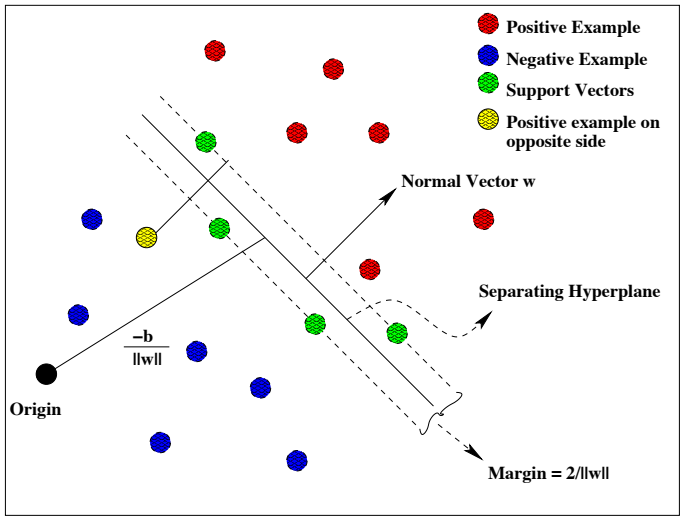

Fig. 6 Linear hyperplane for nonseparable data. Adapted from Burges (A Tutorial on Support Vector Machines)

its corresponding constraint equations employ the dot product of the examples, $\mathbf{x}_i \cdot \mathbf{x}_j$, it was proposed to map the data in a higher dimensional space using a function ϕ s.t. the algorithm would depend only on dot products in the higher space. Next, the existence of a function called *kernel*, dependent on $\mathbf{x}_i$ and $\mathbf{x}_j$, was assumed s.t. the value reported by the kernel was equal to the value resulting from the dot product in the higher space. A mathematical representation of the above concept is -

$$\phi : \mathbf{R}^n \mapsto H \tag{55}$$

$$K(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i) \cdot \phi(\mathbf{x}_j) \tag{56}$$

where H is a higher dimensional space.

This technique drastically reduces the amount of work required while dealing with nonlinear separating hyperplanes, concerning search for appropriate ϕ. Instead, one only works with $K(\mathbf{x}_i, \mathbf{x}_j)$, in place of $\mathbf{x}_i \cdot \mathbf{x}_j$. For classification purpose, where the sign of the function $(\mathbf{x}_{new} \cdot \mathbf{w} + b)$ is evaluated, the formulation employing kernels become:

$$\begin{aligned} f(\mathbf{x}_{new}) &= (\mathbf{w} \cdot \mathbf{x}_{new} + b) \\ &= \textstyle\sum_{i=1}^{N_s} \alpha_i y_i \phi(\mathbf{s}_i) \cdot \phi(\mathbf{x}_{new}) + b \\ &= \textstyle\sum_{i=1}^{N_s} \alpha_i y_i K(\mathbf{s}_i, \mathbf{x}_{new}) + b \end{aligned} \tag{57}$$

where $\mathbf{s}_i$ are the support vectors.

3.7 Ranking Support Vector Machines

4 Description of the dataset & design of experiments

Gujral and MacBeath[17] presents a bigger set of 71 Wnt-related gene expression values for 6 different times points over a range of 24-hour period using qPCR. The changes represent the fold-change in the expression levels of genes in 200 ng/mL *WNT3A*-stimulated HEK 293 cells in time relative to their levels in unstimulated, serum-starved cells at 0-hour. The data are the means of three biological replicates. Only genes whose mean transcript levels changed by more than two-fold at one or more time points were considered significant. Positive (negative) numbers represent up (down) -regulation. Note that green (red) represents activation (repression) in the heat maps of data in Gujral and MacBeath[17]. Figure 3 represents the heat map for the static and time series data respectively. For the case of time series data, interactions among the contributing factors are studied by comparing (1) pairs of fold-changes at single time points and (2) pairs of deviations in fold changes between pairs of time points.

The prodecure begins with the generation of distribution around measurements at single time points with added noise is done to estimate the indices. Two types of distributions are generated; One for the fold changes at single time points and the other for deviations in fold changes that recorded between fold changes at two time points. Then for every gene, there is a vector of values representing fold changes as well as deviations in fold changes for different time points and durations between time points, respectively. Next a listing of all C_k^n combinations for k number of genes from a total of n genes is generated. k is ≥ 2 and $\leq (n-1)$. Each of the combination of order k represents a unique set of interaction between the involved genetic factors. After this, the datasets are combined in a specifed format which go as input as per the requirement of a particular sensitivity analysis method. Thus for each pth combination in C_k^n combinations, the dataset is prepared in the required format from the distributions for two separate cases which have been discussed above. (See .R code in mainScript-1-1.R in google drive and step 4 in figure 7). After the data has been transformed, vectorized programming is employed for density based sensitivity analysis and looping is employed for variance based sensitivity analysis to compute the required sensitivity indices for each of the p combinations. This procedure is done for different kinds of sensitivity analysis methods.

After the above sensitivity indices have been stored for each of the pth combination, the next step in the design of experiment is conducted. Since there is only one recording of sensitivity index per combination, each combination forms a training example which is alloted a training index and the sensitivity indices of the

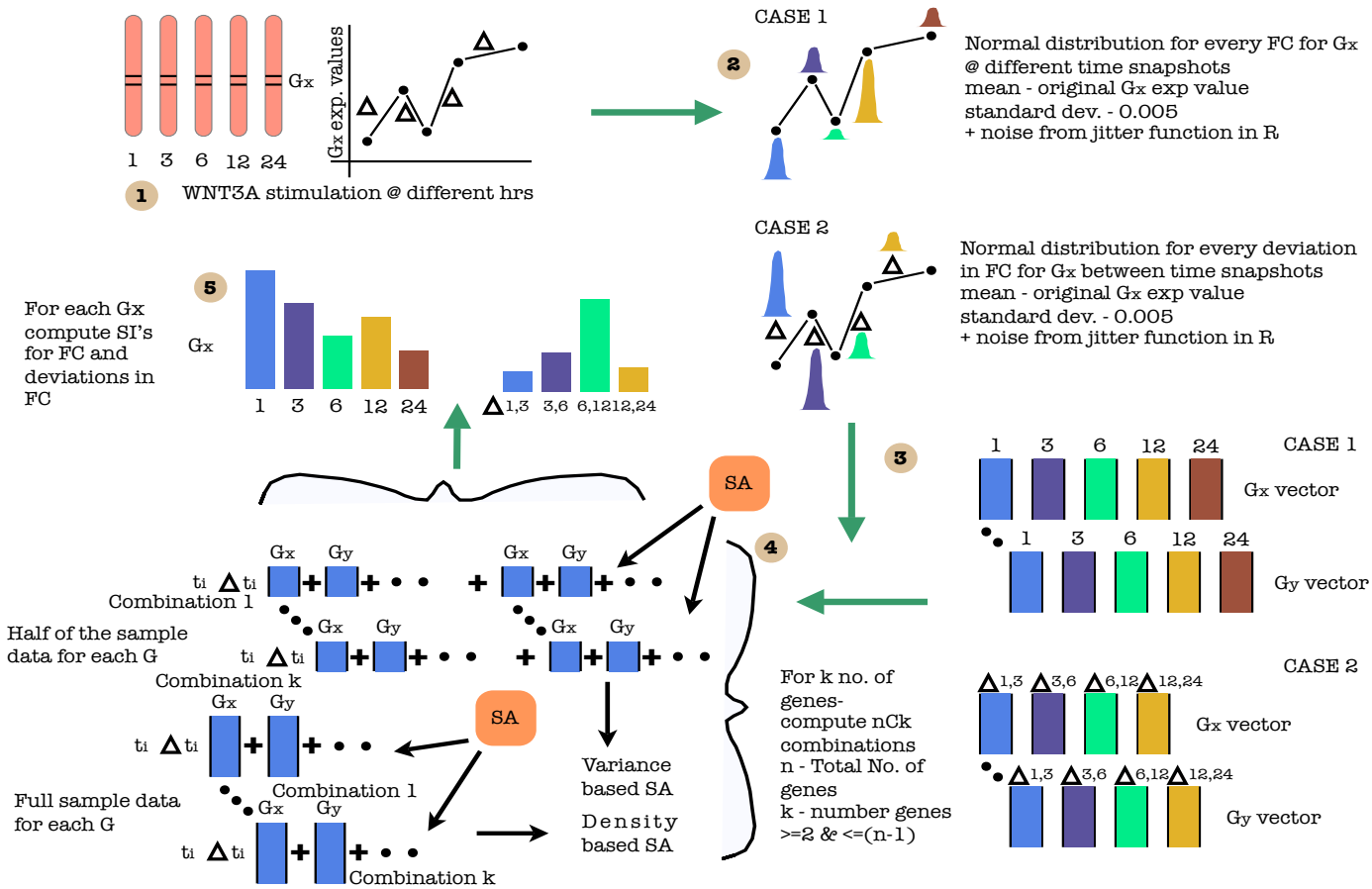

Fig. 7 A cartoon of experimental setup. Step - (1) Time recordings of different gene expression values after WNT3A stimulation at different hrs. Generate deviations in fold changes per gene using the fold change recordings pers time point. (2) Generate of normal distribution for every FC & ΔFC for Gx at & between different time snapshots, respectively. mean - original Gx exp value; standard dev. - 0.005 + noise from jitter function in R (3) Generation of data set for FC & ΔFC. (4) For k number of genes, compute C_k^n combinations and for every SA method, combine data in a specified format for both FC & ΔFC. (5) Compute SI for FC & ΔFC for different.

individual genetic factors form the training example. Thus there are C_k^n training examples for k^{th} order interaction. Using this training set SVM_{learn}^{Rank} (Joachims[80]) is used to generate a model on default value C value of 20. In the current experiment on toy model C value has not been tunned. The training set helps in the generation of the model as the different gene combinations are numbered in order which are used as rank indices. The model is then used to generate score on the observations in the testing set using the $SVM_{classify}^{Rank}$ (Joachims[80]). Note that due to availability of only one example per combination, after the model has been built, the same training data is used as test data to generates the scores. This procedure is executed for each and every sensitivity analysis method. This is followed by sorting of these scores along with the rank indices (i.e the training indices) already assigned to the gene combinations. The end result is a sorted order of the gene combinations based on the ranking score learned by the SVM^{Rank} algorithm. Finally, this entire procedure is computed for sensitivity indices generated for each and every fold change at time point and deviations in fold change at different durations. Observing the changing rank of a particular combination at different times and different time periods will reveal how a combination is behaving. These steps are depicted in figure 8.

– – –WORK IN PROGRESS– – –

5 Results and Discussion

6 General Conclusions

7 Acknowledgement

References

1 R. Sharma, *Drosophila information service*, 1973, **50**, 134–134.

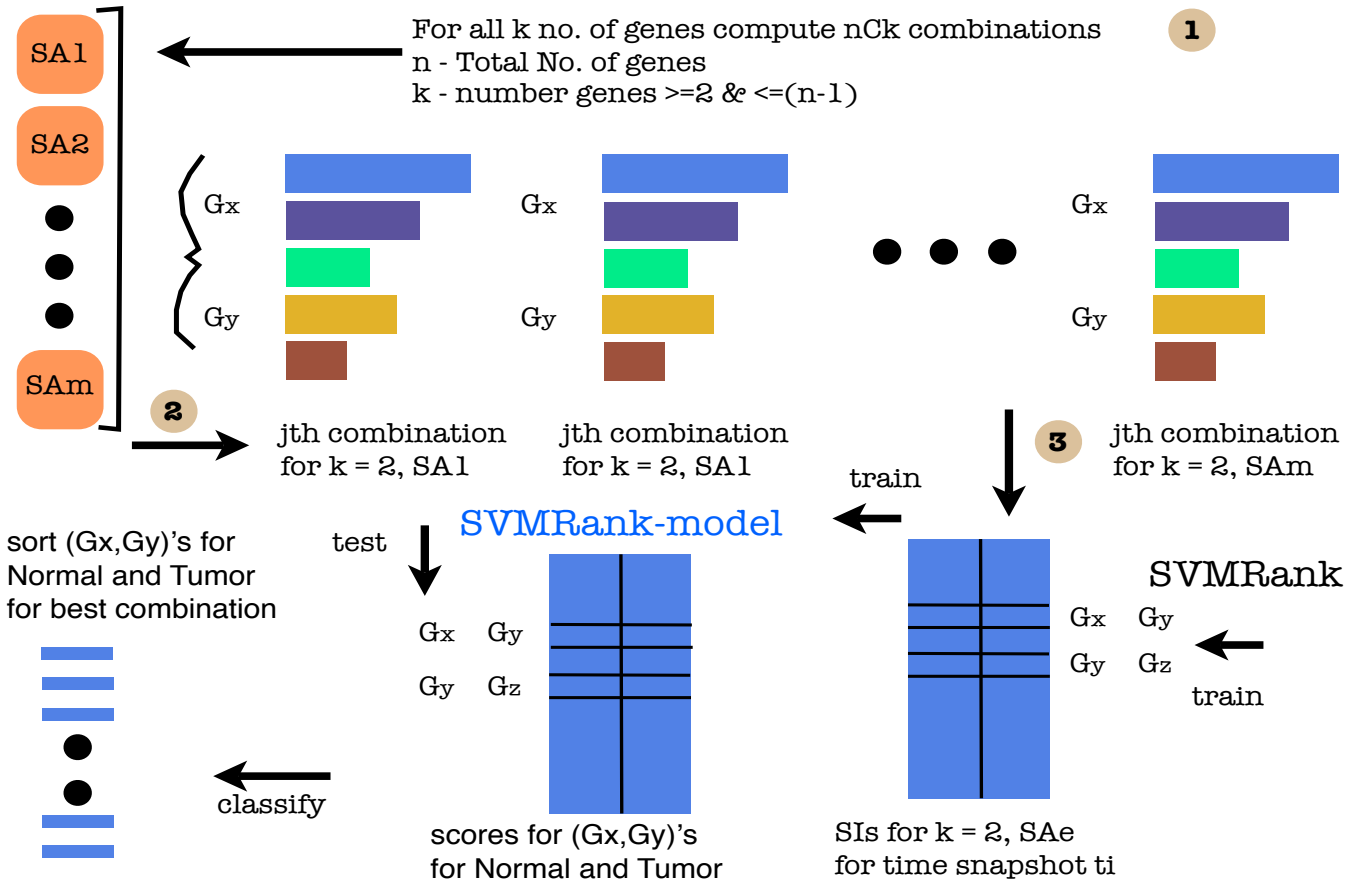

Fig. 8 A cartoon of experimental setup. Step - (1) For k^{th} order genes in C_k^n combinations and for every SA, combine (say for $k = 2$, i.e interaction level 2) SI's of genetic factors form training data. (3) Using indices in (1) SVM_{learn}^{Rank} is employed on training data to generate a model. This model is used to generate a ranking score on the training data again via $SVM_{classify}^{Rank}$. Finally the combinations are ranked based on sorting of training set. This is generated for each and every FC as well as ΔFC. Observing changing ranking in time & time periods reveals how a particular combination is behaving.

2 L. Thorstensen, G. E. Lind, T. Løvig, C. B. Diep, G. I. Meling, T. O. Rognum and R. A. Lothe, *Neoplasia*, 2005, **7**, 99–108.
3 R. Baron and M. Kneissel, *Nature medicine*, 2013, **19**, 179–192.
4 H. Clevers, *Cell*, 2006, **127**, 469–480.
5 S. Sokol, *Wnt Signaling in Embryonic Development*, Elsevier, 2011, vol. 17.
6 D. Pinto, A. Gregorieff, H. Begthel and H. Clevers, *Genes & development*, 2003, **17**, 1709–1713.
7 Z. Zhong, N. J. Ethen and B. O. Williams, *Wiley Interdisciplinary Reviews: Developmental Biology*, 2014, **3**, 489–500.
8 N. Pećina-Šlaus, *Cancer Cell International*, 2010, **10**, 1–5.
9 M. Kahn, *Nature Reviews Drug Discovery*, 2014, **13**, 513–532.
10 K. Garber, *Journal of the National Cancer Institute*, 2009, **101**, 548–550.
11 A. Voronkov and S. Krauss, *Current pharmaceutical design*, 2012, **19**, 634.
12 A. Blagodatski, D. Poteryaev and V. Katanaev, *Mol Cell Ther*, 2014, **2**, 28.
13 J. C. Curtin and M. V. Lorenzi, *Oncotarget*, 2010, **1**, 552.
14 T. P. Rao and M. Kühl, *Circulation research*, 2010, **106**, 1798–1806.
15 J. Yu and D. M. Virshup, *Bioscience reports*, 2014, **34**, 593–607.
16 X. Jiang, J. Tan, J. Li, S. Kivimäe, X. Yang, L. Zhuang, P. L. Lee, M. T. Chan, L. W. Stanton, E. T. Liu *et al.*, *Cancer cell*, 2008, **13**, 529–541.
17 T. S. Gujral and G. MacBeath, *PloS one*, 2010, **5**, e10024.
18 W. Verhaegh, P. Hatzis, H. Clevers and A. van de Stolpe, *Cancer Research, San Antonio Breast Cancer Symposium*, 2011, **71**, 524–525.
19 S. Sinha, *Integr. Biol.*, 2014, **6**, 1034–1048.
20 A. Gregorieff and H. Clevers, *Genes & development*, 2005, **19**, 877–890.
21 I. M. Sobol', *Matematicheskoe Modelirovanie*, 1990, **2**, 112–118.
22 B. Iooss and P. Lemaître, *arXiv preprint arXiv:1404.2405*, 2014.
23 M. D. Morris, *Technometrics*, 1991, **33**, 161–174.
24 H. Moon, A. M. Dean and T. J. Santner, *Technometrics*, 2012, **54**, 376–387.
25 A. Dean and S. Lewis, *Screening: methods for experimentation in industry, drug discovery, and genetics*, Springer Science & Business Media, 2006.
26 T. H. Andres and W. C. Hajas, 1993.
27 B. Bettonvil and J. P. Kleijnen, *European Journal of Operational Research*, 1997, **96**, 180–194.
28 S. C. Cotter, *Biometrika*, 1979, **66**, 317–320.

29 R. Christensen, *Linear models for multivariate, time series, and spatial data*, Springer Science & Business Media, 1991.

30 A. Saltelli, K. Chan and E. Scott, *Willey, New York*, 2000.

31 J. C. Helton and F. J. Davis, *Reliability Engineering & System Safety*, 2003, **81**, 23–69.

32 M. D. McKay, R. J. Beckman and W. J. Conover, *Technometrics*, 1979, **21**, 239–245.

33 T. Homma and A. Saltelli, *Reliability Engineering & System Safety*, 1996, **52**, 1–17.

34 I. M. Sobol, *Mathematics and computers in simulation*, 2001, **55**, 271–280.

35 A. Saltelli, *Computer Physics Communications*, 2002, **145**, 280–297.

36 A. Saltelli, M. Ratto, S. Tarantola and F. Campolongo, *Chemical reviews*, 2005, **105**, 2811–2828.

37 A. Saltelli, M. Ratto, T. Andres, F. Campolongo, J. Cariboni, D. Gatelli, M. Saisana and S. Tarantola, *Global sensitivity analysis: the primer*, John Wiley & Sons, 2008.

38 R. Cukier, C. Fortuin, K. E. Shuler, A. Petschek and J. Schaibly, *The Journal of Chemical Physics*, 1973, **59**, 3873–3878.

39 A. Saltelli, S. Tarantola and K.-S. Chan, *Technometrics*, 1999, **41**, 39–56.

40 S. Tarantola, D. Gatelli and T. A. Mara, *Reliability Engineering & System Safety*, 2006, **91**, 717–727.

41 A. Saltelli, P. Annoni, I. Azzini, F. Campolongo, M. Ratto and S. Tarantola, *Computer Physics Communications*, 2010, **181**, 259–270.

42 A. Janon, T. Klein, A. Lagnoux, M. Nodet and C. Prieur, *ESAIM: Probability and Statistics*, 2014, **18**, 342–364.

43 A. B. Owen, *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, 2013, **23**, 11.

44 J.-Y. Tissot and C. Prieur, *Reliability Engineering & System Safety*, 2012, **107**, 205–213.

45 S. Da Veiga and F. Gamboa, *Journal of Nonparametric Statistics*, 2013, **25**, 573–595.

46 G. Archer, A. Saltelli and I. Sobol, *Journal of Statistical Computation and Simulation*, 1997, **58**, 99–120.

47 S. Tarantola, D. Gatelli, S. Kucherenko, W. Mauntz *et al.*, *Reliability Engineering & System Safety*, 2007, **92**, 957–960.

48 M. J. Jansen, *Computer Physics Communications*, 1999, **117**, 35–43.

49 C. B. Storlie and J. C. Helton, *Reliability Engineering & System Safety*, 2008, **93**, 28–54.

50 S. Da Veiga, F. Wahl and F. Gamboa, *Technometrics*, 2009, **51**, 452–463.

51 G. Li, C. Rosenthal and H. Rabitz, *The Journal of Physical Chemistry A*, 2001, **105**, 7765–7777.

52 K. H. Hajikolaei and G. G. Wang, *Journal of Mechanical Design*, 2014, **136**, 011003.

53 E. Borgonovo, *Reliability Engineering & System Safety*, 2007, **92**, 771–784.

54 I. Sobol and S. Kucherenko, *Mathematics and Computers in Simulation*, 2009, **79**, 3009–3017.

55 J.-C. Fort, T. Klein and N. Rachdi, *arXiv preprint arXiv:1305.2329*, 2013.

56 S. Da Veiga, *Journal of Statistical Computation and Simulation*, 2015, **85**, 1283–1305.

57 G. J. Székely, M. L. Rizzo, N. K. Bakirov *et al.*, *The Annals of Statistics*, 2007, **35**, 2769–2794.

58 A. Gretton, O. Bousquet, A. Smola and B. Schölkopf, Algorithmic learning theory, 2005, pp. 63–77.

59 I. Csiszár *et al.*, *Studia Sci. Math. Hungar.*, 1967, **2**, 299–318.

60 M. Aizerman, E. Braverman and L. Rozonoer, *Automation and Remote Control*, 1964, **25**, 821–837.

61 T. Sumner, E. Shephard and I. Bogle, *Journal of The Royal Society Interface*, 2012, **9**, 2156–2166.

62 Y. Zheng and A. Rundell, *IEE Proceedings-Systems Biology*, 2006, **153**, 201–211.

63 S. Marino, I. B. Hogue, C. J. Ray and D. E. Kirschner, *Journal of theoretical biology*, 2008, **254**, 178–196.

64 L. Goentoro and M. W. Kirschner, *Molecular Cell*, 2009, **36**, 872–884.

65 S. Sobol, IM andKucherenko, *Wilmott Magazine*, 2–7.

66 M. Baucells and E. Borgonovo, *Management Science*, 2013, **59**, 2536–2549.

67 A. Kraskov, H. Stögbauer and P. Grassberger, *Physical review E*, 2004, **69**, 066138.

68 D. Sejdinovic, B. Sriperumbudur, A. Gretton, K. Fukumizu *et al.*, *The Annals of Statistics*, 2013, **41**, 2263–2291.

69 H. Daumé III, *From zero to reproducing kernel hilbert spaces in twelve pages or less*, 2004.

70 F. Riesz, *CR Acad. Sci. Paris*, 1907, **144**, 1409–1411.

71 J. S. Taylor and N. Cristianini, *Properties of Kernels*, Cambridge University Press, 2004.

72 R. Faivre, B. Iooss, S. Mahévas, D. Makowski and H. Monod, *Analyse de sensibilité et exploration de modèles: application aux sciences de la nature et de l'environnement*, Editions Quae, 2013.

73 S. Sinha, *MS Thesis*, 2004.

74 I. K.-U. Bletzinger, *Basic Mathematics*, 2002.

75 C. J. Burges, *Data mining and knowledge discovery*, 1998, **2**, 121–167.

76 N. Cristianini and J. Shawe-Taylor, *An introduction to support vector machines and other kernel-based learning methods*, Cambridge university press, 2000.

77 B. Schölkopf and A. J. Smola, *Learning with kernels: support vector machines, regularization, optimization, and beyond*, MIT press, 2002.

78 V. N. Vapnik and V. Vapnik, *Statistical learning theory*, Wiley New York, 1998, vol. 1.

79 B. E. Boser, I. M. Guyon and V. N. Vapnik, Proceedings of the fifth annual workshop on Computational learning theory, 1992, pp. 144–152.

80 T. Joachims, Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining, 2006, pp. 217–226.