

INForm Demonstrator

1. Introduction

Four examples are illustrated in this **INForm Demonstrator**. The Demonstrator is based on **INForm v3.5**.

All the examples given use literature data. Two of the examples relate to pharmaceuticals – a tablet formulation, and a controlled release tablet formulation. The third is for a hot melt adhesive, and the fourth is an automotive clearcoat. The notes below attempt to explain what the formulator is trying to achieve in each of the three cases, so that you can try to relate these to your own formulation problem.

Although on-line **Help** is available throughout the Demonstrator, the following guide gives a few more tips and pointers about the specific examples given in the Demonstrator, including things which you might want to try for yourself.

2. Loading the Examples

When you launch the Demonstrator, you will see an **Introduction** screen with buttons that include **Create New Task** and **Open Existing Task**. These are not operational in the Demonstrator. In the full version of **INForm**, they allow you to proceed directly to the task that you wish to undertake.

The **New**, **Open**, **Save**, **Save As** and **Recent Tasks** options on the **Task** menu are also not operational in the Demo version. Where features are not available in this Demonstrator, you will be given the message “Feature not enabled in DEMO version”.

To load an example, select **Task | Demos** and choose the example you want to look at. You will be taken immediately to the Set Field Types screen for that example, with the appropriate choices of Ingredient, Processing Condition and Property already made. In the first instance, it is probably sensible to leave the selections as they are. However, subsequently you might want to see what happens if you leave out some of the variables, using the **Not Used** option.

If you want to go back to look at the original data (though you won't be able to change it) then select **Mode | Enter/Edit Data**.

When you click on **OK** at any stage, you will be moved forward automatically to the next step. Clicking on **Cancel** will remove the spreadsheet screens from view, so is probably not advisable!

3. Training the Neural Network

In the **Training** mode, it is useful to select some **Test Data** for validation. Since these literature examples generally used designed experiments, it is not ideal that data points are removed for validation. However, validation of the neural network is the best way to see how predictive it will be.

To remove some data for validation, press the **Test Data** button to get a spreadsheet showing the complete data. Use the **Options** button at the top of this spreadsheet to get the Test Data Options window. **None** will be selected by default; we recommend that you try **Smart** Selection with 10% (the default) of the data records kept for testing.

Press **OK** on the Test Data Options window to complete the selection. You can see which records have been selected as test data by choosing the **Test Data** tab at the top of the Select Test Data spreadsheet. When you have selected the test data, press **OK** to return to the Training window.

On the training screen, you can change the training parameters, if you wish, by pressing the **Parameters** button. In the first instance, you will not need to worry about this, though - just leave it at the default parameters.

Press the **Train** button when you are ready to start training the network.

View Results (from the Training screen) will become active once the model has been developed, and will let you see the Model Statistics, as well as to look at the actual vs predicted values for the test and training data sets. Graphing the actual vs predicted values can indicate visually how good the model is, so is worth trying.

You might also want to see what happens when you change the neural network parameters, or when you withhold different data records for validation. In this second case, you are changing the training set, so that the neural network has different information from which to 'learn'. Consequently, different models may be developed, although they should be fairly similar in form.

Use the **Close** button on the training screen to move forward to the **Consult** screens.

4. Using the Model

The **Consult** screen is where you use the model, for 'what if' investigations and for optimization. Since each of the examples is different, we will discuss the Consult mode separately for each of the three cases, below.

5. Tablet Formulation

The **Tablet** example is taken from Kesavan and Peck (*Proc. 14th Pharm Tech Conference, Barcelona, 1995*). Chapter 6 of *Intelligent Software for Product Formulation* by Rowe and Roberts (Taylor and Francis, 1998) also uses this as its illustrative example. Briefly, this is a tablet formulation consisting of:

- anhydrous caffeine (40% w/w) as a model active
- dicalcium phosphate dihydrate (Ditab) or lactose (44.5-47.5% w/w) as a filler
- polyvinylpyrrolidone (PVP) (2.0 -5.0% w/w) as a binder
- corn starch (10% w/w) as a disintegrant
- magnesium stearate (0.5% w/w) as a lubricant.

Two types of granulation equipment - fluidized bed and high shear mixing - are used, and the binder is added either dry, or as a solution. The amount of caffeine and the percentage of cornstarch were held constant, so the five variables were:

- Diluent (Ditab or lactose)
- Diluent%
- PVP%
- Binder Addition (wet or dry)
- Granulation Equipment (Fluidized Bed or High Shear Mixer)

Properties measured included tablet hardness, tablet friability, tablet thickness and disintegration time. The aim of the optimization is to make hard tablets (which will be robust and will not break up while you are carrying the bottle around in your pocket, for example) that also disintegrate quickly (so that the drug can get to work right away).

Friability has some relationship with hardness, since typically hard tablets are not very friable. Thickness, for the purpose of our study, is pretty unimportant.

Kesavan and Peck carried out a designed experiment, which means that the data were determined at specified points. Therefore, withholding some of the data for training leaves some areas of design space under-represented. You will see the effect of this, in that some of the models train poorly. However, we wanted to use readily accessible literature data, despite this particular limitation for our purposes.

In our case, the variables Diluent, Binder-addn and Gran-equip are all integer values, since they refer to specific classes (e.g. Diluent is Lactose or Ditab). Remember to keep these as integers when you carry out other calculations, since non-integer values are not meaningful for these variables.

To get you started, we suggest that you try the following:

In the **Given** column on the **Properties** side of the screen, fill out the values **10, 2, 2 and 240** for the values of Hardness, Friability, Thickness and Disintegration time respectively. Now, press **Best Match** (at the bottom of the screen) and pick **Properties**. You will see that the Found columns are completed with values that correspond to one of the known formulations. Best Match is simply a retrieval function - it does not use the model but simply looks through existing data for the data record that is closest to our requirements.

Now, press the **Ingredients** button and select **Use Found Values**. This will copy the values in the Found column into the Given column.

Do the same for the **Properties** button.

At this stage, we can try a 'what if' experiment. Change the Given value of the Diluent% to 45, and press the **Predict** button at the right hand side of the screen. The **Found properties** column tells you what the model predicts for this ingredient combination. **3D Graph** can be used to show how one of the properties varies with two ingredients. For example, pick **x** and **y** as **Diluent%** and **PVP%** respectively, and select **z** to be one of the properties. Make sure that there are sensible values (i.e. those lying within the data ranges) in the **Given** or **Found** column so that the 'hidden variables' are treated properly.

The **Optimize** button will take you into the **Optimizer configuration**, where you can set up your objectives for an optimization session, and set up any constraints that you want. Here, the original data had $PVP\% + Diluent\% = 49.5$, so you might want to use that. The Optimize window and the Consult window are coupled, so that when optimization ends, the final optimized values are filled out in the Found columns (for both Ingredients and Properties) of the Consult screen.

6. In Vitro Release

The **Invitro** example looks at how you formulate a tablet with a specific release profile. Models that relate *in vitro* profiles to *in vivo* release exist, so that getting the correct *in vitro* profile is a key step in finding the correct formulation.

The data here are taken from information provided by A I Ware Inc., and were determined by Y Chen and his colleagues. They published their study in the *Journal of Controlled Release* **59** 33-41 (1999), although the data are not actually given in the paper. Chen and colleagues used 10 different formulation variables

- Amount of Polymer A in each tablet
- Amount of Polymer B in each tablet
- Amount of Dextrose
- Amount of Lubricant
- Tablet Weight
- Drug/(Polymer+Drug) ratio
- Polymer A/Polymer B ratio
- Tablet hardness
- Particle size
- % Moisture

Several of these variables are dependent on other variables – e.g. Tablet Weight depends on the amount of other ingredients added (since the amount of drug was a constant 9.6 g), and obviously the Polymer A/Polymer B ratio depends directly on the amounts of Polymer A and Polymer B. We chose to leave these dependent variables out of the model—you can see that by noting that, when you load the data set, they are set as **Not Used**. If you want to try to reproduce the Chen et al paper, you might want to set them to be **Ingredients**.

The outputs are measured releases at different times. Clearly there is error in the measurements, since some of the *in vitro* results suggest a release of over 100%.

When it comes to training, you will see that relatively poor models are developed for these properties, which tend to lie at longer time scales. There are only 24 different experiments, and even for 7 input variables (the number of truly independent variables in this problem) this represents

relatively little data. However, reasonable models can be developed even when 10% of the data are withheld for validation.

In the **Training** screen, you will see that **INForm** recommends a 2-node single hidden layer. This very simple architecture is suggested so that the chances of over-training the network are minimized. Use the Smart selection option to select some Test Data for model validation, and train the model. By looking at the **Model Statistics**, you can see how good the models are. A value of R^2 greater than about 80 is desirable, provided it is supported by a reasonable f-ratio (greater than 4). Many of the models may not be this good. You might want to try withholding different data records, either by using **Smart** selection again, or by selecting records manually. You will probably see that in some cases models are better, while in other cases they are worse.

Once you are reasonably happy with the models, Close the **Training** window and go on to **Consult** mode. Here, the main goal will be to produce a formulation with a specific release profile. If tablet formulation is your field of expertise, you will be able to specify a suitable profile. Otherwise, you could do as we did – look at something where the release was 10% after 1 hour, 20% after 2, 30% after 4, 40% after 6, 50% after 8, 60% after 10, 70% after 12, 80% after 16, 90% after 20 and 100% after 24 hours. If you have included the 'dependent' inputs, e.g. the PolymerA/Polymer B ratio, and the Tablet Weight, you should set up constraints for the optimization to make sure that these are satisfied. Otherwise, the optimization will think that Polymer A/PolymerB ratio is independent of the amount of Polymer A and Polymer B, which is clearly not the case.

The 3D Graphs can be useful to see which of the polymers has the greatest effect at short times, and which is more important for long term release. Remember to make sure that the 'hidden' variables have sensible values when you do the plots!

7. Hot Melt Adhesives

Hot melt adhesives are used in many polymer applications. What is required is a formulation that melts relatively easily, that binds well to substrates, and that gives a strong adhesive bond. And, it is helpful if it is easy to apply, so there are usually requirements for the viscosity, too.

The example given here uses data from Setz and coworkers, reported in the *Journal of Chemometrics* 11 403-418 (1997). Their hot melt adhesive is for bonding to polypropylene, which poses challenges to the formulator – because of its low surface energy, most things don't stick to it. In a typical formulation, oligo(propene) is mixed with SEBS (hydrogenated polystyrene – block polybutadiene – block polystyrene), and a range of tackifiers. In their particular formulation, they used

- iPP10 – an isotactic oligo(propene) with $M_n = 10,000$
- TPE – a hydrogenated polystyrene – block polybutadiene – block polystyrene thermoplastic elastomer (TPE)
- TPEm – like TPE, but with grafted maleic anhydride
- T1 – a hydrocarbon resin tackifier with $M_n 690$
- T2 – a straight mineral oil
-

There is a constraint on the ingredients in this case – the amounts must add to 100%.

29 experiments were reported in their paper. The properties they measured included the lap shear strength (τ_B) and the viscosity, as well as Δ (τ_B).

INForm recommends a 2-node hidden layer for this problem, when 10% of the data is withheld for validation. It is almost impossible to get a good model for Δ (τ_B); perhaps this reflects variation in the experimental data. However, good models are found for the lap shear strength and the viscosity.

You might want to optimize to see if you can find a formulation with high lap shear strength and low viscosity. Remember to add the ingredient constraint, making all three ingredients add to 100%. You can 'trade off' to see if you need to sacrifice viscosity to get high lap shear strength, and vice versa, if you wish. And have a look at the 3D plots, since these show how very non-linear the models are.

8. Automotive Clearcoats

This is another literature example. Kruithof and van den Haak of Akzo Coatings B.V. have reported a study, using statistics, of a clearcoat containing novel monomers. Here, we have used their data, reported in *Journal of Coatings Technology* **62** 47-52 (1990) - but we have treated their data using a neural network. The present note therefore provides useful comparisons with the statistical treatment.

The aim for automotive clearcoats is to increase the solids content, since this means that there is less solvent, in line with environmental pressures. Adding monomers with linear flexible bulky groups increases the solids content, but reduces the hardness of the coating, and this is not desirable. Monomers with rigid bulky groups (rather than flexible ones) are expected to improve the solids content, but without sacrificing the hardness. Kruithof and van den Haak looked at four possible monomers.

For each monomer, there were 20 experiments, varying the film thickness, the percent of novel monomer, and the percent of melamine formaldehyde crosslinker. Because of the limitations of the statistical package, only three variables were used - neural nets can of course cope with many more variables.

Kruithof and van den Haak found that, of the 4 monomers they considered, TMCMA led to the highest hardness, so this example uses the data just for that monomer. Consequently, there are three input variables - Monomer %, MF (melamine formaldehyde) %, and film thickness. Two properties - the solids content, and the hardness, were measured.

In their data, one point gives a higher value than usual for solids content. We have assumed that this is a valid point, but it may be a typographical error. If the latter case is true, then we might expect to develop relatively poor models.

INForm suggested a default architecture with a single hidden layer containing 3 nodes. Separate models were developed for Solids Content and for Knoop Hardness.

Because this is a designed experiment, removing data for validation is not ideal. However, it is worth doing anyway, so that you have an idea of the predictive capabilities of the model. You may find that you get fairly poor models, as shown by the MS Error and the Model Statistics. (Look at these by pressing the **View Results** button.) If you get a poor model, try aborting the training, and picking a different random selection of withheld points, using the **Set Test Data** button. You can also change the network parameters - the Learning Rate, Momentum and Random Seed, if you are feeling ambitious.

The ANOVA statistics generally show that the model captures only about 60-70% of the variation in the data. This might indicate that other factors, which have not been measured, are having an effect.

By exploring the 3D graphs, you can see that the behaviour is quite non-linear for both properties - and, interestingly, hardness is not linear with film thickness.

9. And in Conclusion

Remember, there is on-line help available throughout the Demo, to show you what the different buttons are for.

We hope that you have found this **INForm Demonstrator** useful. Please refer any queries (or provide any feedback) to

Intelligensys Ltd
Springboard Business Centre
Ellerbeck Way
Stokesley, Cleveland TS9 5JZ
e-mail: postmaster@intelligensys.co.uk